The Thesis committee for Kyle Anthony Schroeder

Certifies that this is the approved version of the following thesis:

On the Use of Generalized Force Data for Kinematically Controlled Manipulators

APPROVED BY

SUPERVISING COMMITTEE:

Supervisor: _____
Sheldon Landsberger

Co-Supervisor: _____
Mitch Pryor

On the Use of Generalized Force Data for Kinematically Controlled Manipulators

by

Kyle Anthony Schroeder, B.S.E.

Thesis

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

Master of Science in Engineering

The University of Texas at Austin

December, 2010

Dedication

To my grandfather, James Makley, whose passion for learning has inspired, motivated, and excited me. And to my parents who encourage and nurture my curiosity.

Acknowledgments

I would like to thank Mitch Pryor for his guidance and Sheldon Landsberger for his aid and assistance. I would also like to thank Troy Harden at Los Alamos National Laboratory for his help.

This work was partially funded by the National Nuclear Security Administration under the University Research Program in Robotics, grant #DE-FG52-06NA25591 and by Los Alamos National Labs under project # 79506-001-10.

Abstract

On the Use of Generalized Force Data for Kinematically Controlled Manipulators

Kyle Anthony Schroeder, M.S.E.

The University of Austin at Texas, 2010

Supervisor: Sheldon Landsberger

Co-Supervisor: Mitch Pryor

The Department of Energy national laboratories, like Los Alamos National Lab or Sandia National Lab, perform work on radioactive and chemically dangerous materials. Gloveboxes are often used to shield workers from these hazards, but they cannot completely eliminate the danger and often create new safety concerns due reduced operator dexterity and ergonomic posture. When feasible, robots can be employed to remove the human from the radioactive hazard; allowing them to analyze the situation and make decisions remotely.

Force sensor data from the manipulator can be used by to simplify the control of these remote systems as well as make them more robust. Much research has been done to

develop force and torque control algorithms to introduce compliance or detect collisions. Many of these algorithms are very complicated and currently only implemented in research institutions on torque-controlled manipulators. The literature review discusses many such controllers which have been developed and/or demonstrated. This thesis reviews, develops and demonstrates several beneficial algorithms which can be implemented on *commercially-available* kinematically-controlled robots using *commercially-available* sensors with a reasonable investment of time.

Force data is used to improve safety and manage contact forces while kinematically controlling the robot, as well as improve the world model. Safety is improved by detecting anomalous and/or excessive forces during operation. Environmental modeling data is inferred from position and/or force data. A six-axis sensor and a joint torque sensors on 2 7DOF manipulators are used to demonstrate the proposed algorithms in two DOE relevant applications: remotely opening an incompletely modeled cabinet door and moving a robot in a confined space.

Table of Contents

List of Tables

List of Figures

1 Introduction

1.1 Hazardous environments

The Department of Energy (DOE) and National Nuclear Security Administration (NNSA) operate nuclear laboratories around the country. Some of the work done in these labs requires the physical manipulation of radioactive materials which emit dangerous levels of radiation. In order to protect workers, exposure is kept As Low As Reasonably Achievable (known as the ALARA principle). Several forms of protection are used to keep workers safe. Health physicists monitor the dose received by employees and take action if the dose becomes too high. Barriers – or shielding – can reduce dosage while permitting the work to be performed. One of the most common barriers is the glovebox.

Figure 1-1: Example glovebox [Merrick, 2010]

Gloveboxes completely surround a volume to protect what is outside from what is inside or *vice versa*. The box is often made of stainless steel with shielded glass panels to allow workers to see into the box. There are also several ports where gloves can be attached. Although gloveboxes provide some protection, dangers for workers in these environments still persist. These include

- long-term, low-level exposure to hazardous substances,
- contamination due to breaches in the shielding, particularly in the gloves, ports, and other interfaces, and
- ergonomic and other injuries that can occur due to the awkward workspace and reduced dexterous operability when handling objectives through the gloves.

The glovebox may contain radioactive or chemically hazardous materials which pose a threat to the employee. Tools and objects in the glovebox might puncture the gloves or a worker's skin. Such a breach may contaminate the worker and/or the surrounding area.

1.2 Robotics in hazardous environments

Robots in the glovebox allow the worker to be moved away from the radiation and ergonomic hazards. More shielding can be used to protect the employee because the robot can be operated remotely. The robot controller can be designed to allow people of different sizes and statures to work more comfortably. In some cases, such as applications using the Mighty Mouse manipulator presented below, it is impossible to properly protect the human and allow them to do the work. The use of a robot is necessary. However, using robots has its own unique challenges. Major considerations when deploying robots include

- **Hazardous conditions** – Robots are more resistant to most of the conditions which endanger humans but they are still susceptible to damage from radiation and chemical hazards.
- **Confined workspaces** – Robots may require more space than a human operator to move in order to dexterously manipulate objects in their environment.
- **Unstructured environments** – Robots can't collect and process data as quickly as humans. They are easily disoriented or may cause damage in a changing environment.
- **Limited functionality** – Robotic systems cannot typically perform the spectrum of tasks that an operator can or in an environment with as much uncertainty as an operator can tolerate.

In this thesis, a robot's ability to resist hazardous chemicals and radiation will not be addressed. Work is being done in the University of Texas at Austin Nuclear Robotics Group to better understand and improve these limits. This thesis will instead focus on how force and torque data collected from commercially available sensors can improve the operational capability of kinematically (i.e. *position, velocity, etc.)* controlled manipulators.

1.3 Current manipulation techniques

Many manipulators used in gloveboxes are tele-operated or perform moves which are preprogrammed. The environments are well-structured and the tasks consistent and repetitive so the robot can easily accomplish them. If the environment changes, the pre-programmed moves may not lead to the robot successfully lifting its target object or may cause it to collide with unmodeled objects. This section introduces several prominent automation efforts within the DOE complex to provide both context and a baseline for this research effort.

1.3.1 ARIES

The Advanced Recovery and Integrated Extraction System (ARIES) project has automated several tasks in the disposal of surplus weapons-usable fissile material. [Turner and Lloyd, 2008] The Pit Disassembly (PDIS) process requires more oversight because the robot operates within the confines of a glovebox. Its task space is shared by a robotic lathe and a tool changer. Because it works in an enclosed, dynamic environment, the PDIS robot operates in a semi-autonomous mode. A human oversees the operation of the robot, choosing its tasks and coordinating its movements with the other robots in the glove box. [McKee, 2008]

1.3.2 Mighty Mouse (M2) robot

As an example, at White Sands Missile Range, a robot was used in 2005 to accomplish an important task too dangerous for humans. A Cobalt-60 source became lodged in a transport tube. The sample needed to be dislodged and returned to its storage location. The source was strong enough to "kill a man in half a minute". [Sandia National Laboratories, 2005] To perform the task, the M2, or Mighty Mouse robot (Figure 1-2) was deployed and withstood enough radiation to kill 40 men. The robot was completely tele-operated by operators located a safe distance from the source. The humans were removed from the danger but the robot relied heavily on them to choose tasks, process feedback, and for guidance in every manner. It is also worth noting the robot suffered serious radiological and mechanical damage while completing the task.

Figure 1-2: Might Mouse (M2) robot [Sandia National Laboratories, 2005]

1.3.3 Sphere Cleanout

The Spherical Vessel Decontamination (SVD) project is a program tasked with cleaning and decontaminating spherical vessels which were previously used in nuclear experiments. They contain debris of unknown size and composition and contain unknown radiation hazards. Decontamination of the spheres requires that all debris be removed and the inside be cleaned with a steel brush. In order to ensure success, it is swabbed to test for remaining contamination. Because of the radiation and the confined space classification of the spheres, human entry would be very expensive and dangerous; requiring long and expensive confined space entry procedures as well as protective clothing and/or shielding. Cleaning the sphere by hand from outside is difficult and potentially dangerous including the risk of ergonomic injuries.

Figure 1-3: RRC-K1207i inside a sphere

A robot will be used to improve efficiency, reduce dosage, and improve safety. The robot will augment the human operator's capabilities; allowing the operator to make all the decisions, manually controlling the robot. The robot is resistant to the radiation dose and can easily reach inside the sphere, accessing all parts of the sphere and efficiently removing debris which too heavy for someone reaching through a glovebox port. The robot is a 7 Degree-of-Freedom (DOF) Robotics Research Corporation RRC-K1207i arm. It will be used to clean the inside of these 6-foot diameter spheres. Different tools have been designed to accomplish the tasks required to decontaminate the sphere. The arm came supplied with a controller which provides motion planning capabilities so the operator can easily perform repeated tasks or those which require a high level of precision. The robot is shown in the sphere in Figure 1-3.

In each of the examples, force and torque data could be used to improve operational safety and reduce the burden on the operator.

- In the ARIES system, force and torque data could be used to identify collisions and assist in manipulation.
- A six-axis force sensor mounted at the wrist of the Mighty Mouse robot could have verified contact with the environment, guided tool usage, and prevented excessive forces.
- Joint torque sensors in the RRC arm can be used to detect collisions, as will be demonstrated in Chapter 6, and may even be used to determine and control contact force during manipulation.

1.4 Thesis objective

This thesis will demonstrate the use of force and torque data in order to improve the ability of *commercially-available* serial manipulators to operate in hazardous environments. Using hardware of the types described in Chapter 2, robot functionality will be improved in three ways:

- Improved Safety
- Management of contact forces
- On-line augmentation of the world model

Algorithms will be presented which use sensor-agnostic force data to improve the performance of two different robots. Two types of sensors, a six-axis force sensor and joint torque sensors, will be used for two demonstrations. In the first, the six-axis sensor is used to gather data useful for the opening of a cabinet door. The second demonstrates the ability to detect collisions using joint torque sensors.

Much of the applied research in this area focuses on directly integrating force data into the low-level control algorithms to perform a variety of proposed *dynamic control* schemes that are found in the literature and some research robotics. The commercial

viability of these systems has proven elusive as they represent a promising, but complex and unproven technology. This effort bridges the gap between pure kinematic and dynamic control schemes. By introducing beneficial (and deployable) uses of this data on commercially available systems, this effort provides a bridging operational capability between these two control paradigms.

2 Available Hardware

2.1 Chapter Overview

As is clear in the examples above, this work focuses on serial manipulators. Additional considerations, such as load distribution, must be made for parallel manipulators. For work done at the Robotics Research Group (RRG) on parallel manipulators, refer to Freeman and Tesar [1988], Hudgens and Tesar [1988], and Craver and Tesar [1990]. A brief overview of available sensor technologies in commercial serial manipulators is given. A framework of force data is developed in conjunction with available hardware for sensing forces. The available sensors will be considered while developing the framework in order to maintain the focus on development for existing commercial technology.

2.2 Manipulators

Serial manipulators commonly range from three to seven DOF. When the task dictates limited motion in well-defined frames, a manipulator with fewer DOF may be used. Where tasks change frequently or dexterity is required, manipulators with higher DOF can be easier to program and maintain than machines designed for a particular task. Generally, the algorithms presented in this work can be applied to systems with less than six DOF but the focus shall be on manipulators with at least six.

Manipulators having more than six DOF are considered redundant manipulators. A redundant manipulator can achieve a given set of end effector coordinates, position and

orientation, in an infinite number of joint configurations. The system inverse kinematics has an infinite number of solutions. Redundancy has been used by the RRG for obstacle avoidance [Harden, 2002] and for optimization [Hooper, 1994]. These algorithms use criteria for choosing one solution from the infinite set. The optimization criteria may be used to reduce joint speed, increase force transmission, or keep joints positions as near to the center of their range as possible, but any other criteria may also be chosen as the designer sees fit.

When using force sensing algorithms the redundancy may prove to be important. When the manipulator encounters an unexpected collision, there are techniques for responding safely, but none which determines the exact location of the collision for future avoidance. It may be possible to exploit redundancies in order to change the joint positions while maintaining the contact in order to better identify the particular location of the collision in space. Additionally, redundancy is of consequence to the calculation of joint torques due to manipulator loading. The joint torques due to loading must be calculated from a pseudo-inverse of the transpose of the Jacobian instead of the true inverse.

2.3 Commercially-Available Manipulator Controllers

Manipulators can be purchased with different controllers and control systems. Most companies offer a variety of systems to fit the needs of the customer. Available control systems range from the basic; offering forward and inverse kinematic tele-operation and pre-programmed path execution, to the complex which incorporate vision systems and force sensors. Vision sensors are used for target identification and tracking tasks including bin picking and tracking objects on a conveyor belt. [RobotWorx, 2004] Force control

systems are widely available for manufacturing tasks like grinding, machining, and finishing. These systems allow the robot to respond to differences in part thickness or position by detecting the force during machining. These force controllers help prevent the manipulator from binding, stalling, or damaging the tool. [Brumson, 2007]

Some packaged control systems, such as the EPSON RC520 and RC420 controllers, are capable of detecting irregularities in motor torque. [Epson, 2010] Others, like the Fanuc CollisionGuard system, provide collision detection for the robot, but it is unclear from published information how the system functions. Because the EPSON system monitors motor torques for irregularities, they have the potential to identify collisions with any part of the robot. However, the gearing of the robot may also make it difficult to detect collisions until the contact force is very high or miss collisions entirely if contact occurs parallel to the joint's axis.

2.4 Compliant Devices and Collision Sensors

Compliant devices are designed to passively compensate for positional errors during assembly or manipulation. The Schunk Compensation Unit, for example, allows positional compensation in two translational directions up to 5mm. The flexibility can be set for particular circumstances using adjustable internal springs. Figure 2-1 shows a compliant device being used to compensate for error in the location of the hole while inserting a peg [RAD, 2010]. If the geometry of the part and hole are not properly designed, the compliant device may be ineffective. The errors in insertion must create a force in a compliant direction. In Figure 2-1, the part is beveled so the force of contact has a component parallel to the compliant direction of the device, as indicated by the black

arrows. If the part were flat, the force would be directed parallel to the centerline of the part, perpendicular to the compliant direction, thus rendering the compliant device ineffective. Alignment device designs vary in the axes of compliance and can be chosen to meet specific task needs. Figure 2-1 shows different axes of compliance for compliant devices and collision sensors. Compliant devices can be fitted with accessories such as tool changers or collision sensors making them useful in a variety of applications.

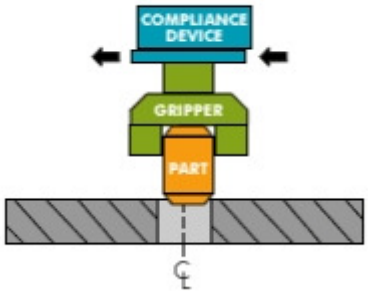
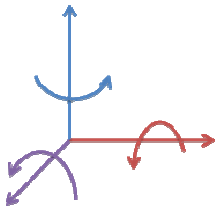
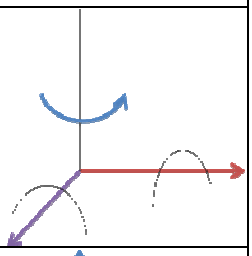
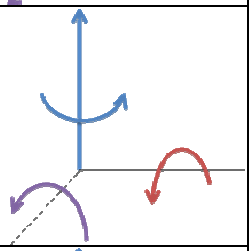
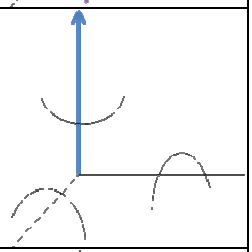
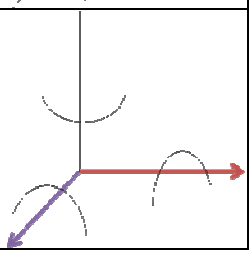


Figure 2-1: Compliant Device

Collision sensors and load limiters are similar to compliance devices except they shut down the robot when the positional error is too large. These switches operate by opening or closing a contact which is held by a spring or pneumatic device until the force is large enough. Because the speed and acceleration of the links downstream of the sensor generate a force on the sensor, if the manipulator is moved too quickly, the sensors may falsely indicate a collision. They can be simple, cost-effective solutions for manipulators where the speeds and accelerations will be low or the maximum forces due to free-space motion can be predicted. Many mechanical collision switches can be adjusted manually while pneumatic switches are adjustable and resettable online by air pressure. A drawback of pneumatic systems is that they require an air source and associated plumbing.

Table 2-1: Compliant device axes of compliance

Example	Compliant Axes
IPR Remote Center Compliance Device	
RAD Compliance Devices	
RAD Collision Sensor	
IPR Z-Axis Compliance Device	
IPR Lateral Compliance Device	

The compliant hardware options described above can be cost-effective, efficient solutions to position error problems. However, these solutions still rely on the existence of a world model of high fidelity. They cannot be used to locate a hole for which the position error is larger than the compliance of the device. The success of the compliant device is also

dependent on the shape of the two pieces being aligned. The collision detection switches can only be used to detect collisions down-stream of the sensor. If placed nearer the base, the sensor must resist forces due to the motion of the manipulator which decreases sensor sensitivity and manipulator speeds and accelerations.

2.5 Force Sensors

Force sensors can be used to detect collisions and provide data which can augment the world model. Force sensors offer greater flexibility by providing constant, continuous force feedback which the computer can analyze. Comparatively, the compliant systems only provide visual feedback and the collision switches offer only an indication that a certain threshold force has been exceeded.

In order to provide useful analysis of the data and identify the algorithms which are best implemented with a given set of sensors, it is necessary to classify the sensors and their feedback type. The details of implementation of each algorithm will vary for each sensor. Different sensors require different hardware and signal processing to interpret the electronic signals from the transducer. There exist several different interfaces including Data Acquisition (DAQ), NET F/T, and CAN. Despite differences in protocol, hardware, and controllers, the sensors measure forces and/or torques in a given set of directions. The data can be broken down into pure forces and moments and each sensor provides a different sub-set of the six general forces along (pure forces) or about (moments) the three primary axes.

2.5.1 Six-axis force sensors

The six-axis sensor measures pure forces along all three axes and torques about each axis as shown in Table 2-2. Each figure in the table represents all three axes with rotations about each axis. For the six-axis sensor, all of the arrows are bold to indicate that all six forces are being measured by this sensor. This sensor provides what will be called a Six-Axis Force because it measures the full complement of forces at and about a particular point.

Six-axis force sensors come in a wide variety of sizes from less than 2cm to over 30cm in diameter and their load capacities go up to 40,000 Newtons and 6,000 Newton-meters. The wide range of sizes makes them practical in applications of many magnitudes, from circuit board assembly to car crushing. Resolutions vary depending on the model and communication protocol. Resolution also depends on the axis of measurement. ATI 16-bit DAQ and Net F/T sensors are available with resolutions down to 1/682N.

These sensors are often wrist-mounted on the robot; between the last joint and the robot end effector. They can be used to control the contact force between the robot and its environment, detect tool collisions, assess loads, and offer tactile world model augmentation. Some of these capabilities will be discussed in greater detail in Chapter 5.

2.5.2 Torque sensors

Joint torque sensors measure the joint torque about the joint axis of rotation. In Denavit-Hartenburg notation (Section 4.3), this is the *z*-axis of the local joint coordinate frame. For this report, the data from this type of sensor will be classified as a torque about

the *z*-axis as illustrated in Table 2-2. Joint torque sensors are able to provide feedback pertinent to the entire robot such as collision forces or end effector forces. Because the geometry and current configuration of the entire robot dictate the joint torque given a particular configuration and load, it can be difficult to identify a contact force using joint torque. For certain configurations of the manipulator it may be impossible to detect forces in a particular direction.

The use of joint torque sensors for collision detection and dynamic control algorithms(impedance, admittance, etc) will be discussed in Chapter 3. The calculation of end-effector force and a model for manipulator dynamics will be discussed in the manipulator modeling section. Their use for collision detection will be demonstrated in the algorithm demonstration section.

2.5.3 Contact skins

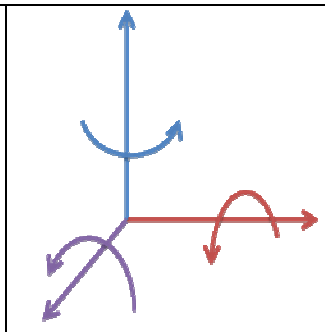
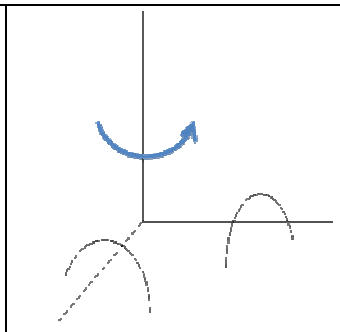
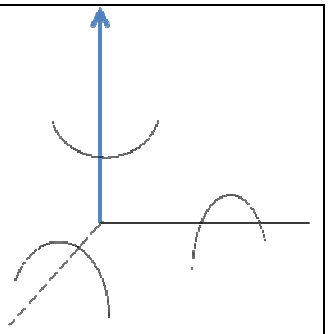
Human skin is capable of detecting many features of its environment through contact. Besides sensing temperature, skin can detect the pressure and point of contact with its environment. Because the human body is covered in this highly-powerful sensory material, the perceptive human can detect collisions and relative forces at nearly any point on the body. The ability of skin to provide so much information over a large area makes sensory skins an attractive sensor to implement on robotic manipulators.

There are few sensory skins available commercially. Several types of skin are under development. Some use capacitive sensors or arrays of pressure sensors. Others are using a new material called a Quantum Tunneling Composite (QTC). [Marks, 2010] All the skins

detect contact and most of the advanced skins can detect the force applied. Most of the skins being pursued today are also capable of identifying the location of contact. There is the potential to develop skins which are sensitive to chemicals like robotic noses distributed across a surface. [Marks, 2010] Radioactively sensitive skins may also be useful.

As a collision detection technique, skins are still expensive, use sensitive materials, or are hard to deploy on highly flexible manipulators. Further testing and more research implementation may reduce their cost. Material advances may allow the skins to be deployed in hazardous environments. Some skins are reducing the number of wires, making them more viable for highly-dexterous manipulators.

Table 2-2: Force sensor sensing axes

Forces:			
Sensor:	Six-Axis Sensor	Joint Torque	Contact Skin

2.6 Chapter summary

Compliant devices and collision switches are available in the form of hardware but these devices are limited in their range of compliance, ability to provide useful contact

information, and diversity of applications. Force sensors provide more diversity but require more computation to use raw data in useful algorithms. Six-axis sensors can measure forces and torques in each of the six directions while torque sensors and contact skins can measure only one direction each. The distribution of the torque sensors and the skins across the length of the robot give them the advantage of detecting contact along a larger portion of the robot. The use of the data gathered by these sensors will be discussed in further detail in Chapters 3 and 5.

3 Literature Review

3.1 Chapter overview

While attempting to identify uses of force data for collision detection and world model augmentation, it is prudent to mention the fundamental control laws developed for serial manipulators using force and torque feedback. Much work has been done in the area of force and torque control such as impedance and compliance controllers. Collision detection algorithms include both joint torque and six-axis force sensor algorithms. The research done in the area of world modeling based on force data will also be presented.

3.2 Force and torque controllers

The simplest method for controlling force during interaction between robot and environment is through passive stiffness control. Passive stiffness makes use of the natural compliance of the manipulator, tool, or environment to limit force of interaction. The compliant devices discussed in the previous chapter offer passive compliance. As another example, a tool mounted on a spring, like the brush used in the Containment Vessel Disposition project to be discussed later, provides passive stiffness control. Positional errors translate into smaller changes in force when the system is less stiff but positional accuracy and force transmission are also more difficult to achieve with a compliant manipulator. Passive stiffness is a viable solution for many situations including tele-operated manipulators where the operator has a good view of the complaint device and can determine that the force applied is within a reasonable and safe range.

Active force and torque controllers use force, torque, position, velocity, and acceleration data to control robot interactions with the environment. The data used and the method of controlling the robot varies from algorithm to algorithm. Most of the algorithms measure force or position in order to modify the force or position error, but others, such as the hybrid impedance algorithm, measure force and modify the velocity and force errors. A good overview of the fundamental force control algorithms is given by Zeng and Hemami. [1997] Table I on page 478 of Zeng and Hemami [1997] provides a breakdown of each algorithm's workspace, measured and modified variables, and modulated objective.

Table 3-1: Force control summary table [Zeng 1997]

Algorithm classification		Workspace	Measured variables	Modified variables	Modulated objectives
Active stiffness control	1. Version one	Joint space	Position force	Joint displacement, contact force	Joint stiffness matrix
	2. Version two	Task space[a]		Position error, contact force	Stiffness matrix
Impedance control	1. Basic impedance control	Task Space	Position, velocity, force,	Position and velocity error, contact force	Impedance
	2. Position-based impedance control			Modified desired trajectory, contact force	
Admittance control			force	force error	Admittance
Hybrid control	1. Hybrid position/force	$\{P\}^b$	Position	Position error	Position
		$\{F\}^c$	Force	Force error	Force
	2. Hybrid impedance	$\{P\}$	Force	Velocity error	$Z_{mp}{}^d$
		$\{F\}$		force error	$Z_{mf}{}^e$
Explicit force control	*PI*, *PD*, *PID*, etc.	Task space	Force	Force error	Desired force F_D
Implicit force control		Task space	Position	Position error	Predefined stiffness

[a] Task space = $\{P\} \oplus \{F\}$.
[b] $\{P\}$ is Position subspace.
[c] $\{F\}$ is Force subspace.
[d] $\{Z_{mp}\}$ is the impedance expressed in Position subspace.
[e] $\{Z_f\}$ is he impedance expressed in Force subspace.

Advanced algorithms build on these fundamental algorithms to make the systems more robust and reduce model and output errors. Zeng and Hemami [1997] also provide a good classification and overview of advanced control algorithms.

3.3 Collision detection

Force control algorithms are primarily designed for tasks requiring controlled forces during interaction with the environment such as cutting or polishing. In some cases contact with the environment is unplanned and undesirable. In static environments, collisions can usually be avoided using model-based collision detection algorithms. Other collision detection methods use information from force, torque, and/or position sensors, models, or switches. Whereas most force controllers use force data to accomplish a primary interaction objective, collision detection algorithms monitor forces and identify forces which represent an unintended interaction force.

3.3.1 Collision detection hardware

Collision detection can be achieved passively by collision detection switches such as those discussed in the previous chapter. Commonly they are mounted at the end-effector and only detect collisions with the tool. These sensors allow collision detection that is very easy to implement, but lacks the flexibility of other systems – e.g. tolerances must be set and adjusted manually or pneumatically. See the section on collision detection switches in the previous chapter for further information.

3.3.2 Model based collision detection

Collision detection is also performed with models of the physical system. OSCAR[1] implements collision detection algorithms developed at the RRG [Harden & Tesar, 2002] [Swint & Tesar, 2004]. These algorithms use three dimensional models of the robot and environment and calculate several criteria used to quantify the proximities (in time and distance) to collisions. They can also be used for obstacle avoidance; planning and executing paths that avoid obstacles. A major strength of these algorithms is that they do not require feedback from a force or collision sensor, reducing system cost. However, the environment must be accurately modeled. Inaccuracies can lead to an unnecessarily reduced workspace (over-bounded inaccuracies) or to collisions (under-bounded inaccuracies).

3.3.3 Visual feedback methods

Vision sensors can also provide data used for collision detection for safe interaction and obstacle-free motion. Numerous methods have been demonstrated including human-robot interaction with image-based collision detection [Ebert, 2002], robotic system motion control with vision guidance [Borangiu, 2006], real-time collision avoidance using light-weight, high-speed vision systems [Morikawa, 2007], and online task and vision based trajectory generation [Eitner, 2008] among many others.

[1] Operational Software Components for Advanced Robotics (OSCAR) is a series of C++ libraries developed at the University of Texas at Austin. [Kapoor, 1996] OSCAR is designed to facilitate robotics research by generalizing robotics-related functionality in a framework quickly adaptable to different hardware systems.

3.3.4 Collision detection with six-axis sensors

Collision detection can also be performed using six-axis force sensors. Methods using the six-axis sensor include using a single sensor at the end effector [Zheng, 1986] [Uchiyama, 1989] and a pair of sensors; one at the base, the other at the end effector [Lu, 2005]. Using the Force/Torque (F/T) sensor at the end effector, only collisions with the end effector can be detected. Using Lu's method, the collision can be identified anywhere along the length of the manipulator. When the end effector is also in contact with the environment, the location of contact and force exerted by the collision can also be determined.

3.3.5 Collision detection with joint torque sensors

Joint sensor based methods of collision detection have also been demonstrated. By detecting at each joint along the length of the robot, these methods allow detection of collisions at more points along the robot. Their ability to detect collisions is limited by the configuration of the robot. For example, forces exerted on the robot which are parallel to the axis of a joint will not be detected by that joint. Both force sensor and sensorless methods have been presented.

3.3.5.1 Sensorless detection

Several methods have been examined to detect collisions without directly sensing the forces. De Luca [2005] demonstrates a sensorless method which detects collisions only monitoring joint positions and velocities. Je [2009] uses joint currents and control input values to detect collisions. Both of these algorithms detect the existence of collisions and

can determine that a collision occurred on a rigid body outside (meaning towards the end effector, not the base) of a given joint axis.

3.3.5.2 Torque sensing

The disturbance torque method can use joint torque sensors or calculate the torque from the joint position, velocity, and acceleration from the manipulator dynamics. Ralph [1995] demonstrates the disturbance torque method and shows that it can be used to identify the location of collision for planar manipulators.

3.3.5.3 Impedance control with collision detection

Impedance control has been demonstrated for collision detection by Matsumoto [2001] and Morinaga [2003] and others. Their algorithm detects collisions between the end effector and the environment by detecting a difference between the modeled and detected torques while using impedance control. The advantage of collision detection using impedance control is that the controller is immediately prepared to navigate over the surface of the obstacle.

3.4 Chapter summary

Several methods of force control have been researched. The current level of research in this area is quite developed. Several methods for collision detection have also been demonstrated in research environments. Cutting-edge research techniques are too complicated for implementation in most industrial applications. It is one of the objectives of this effort to advance the state-of-the-art in advanced but deployable systems. A simple

collision detection scheme using a model of the manipulator's dynamics will be discussed in Chapter 5 and demonstrated in Chapter 6.

4 Manipulator Modeling

4.1 Chapter overview

An understanding of the kinematic and dynamic models of the manipulator is necessary in order to develop force and torque algorithms for control. The kinematic and dynamic calculations are done using the kinematics of robots based on matrices developed by Denavit and Hartenberg [1955]. Notation, frame assignments, and transformations are discussed. Manipulator dynamics are discussed. For more information on the topics discussed in this chapter, see Craig [2005].

4.2 Geometric representations

4.2.1 Vectors

Vectors point from the origin of a reference frame to a point of interest. For Denavit-Hartenberg (D-H) parameters, the point of interest is the origin of another frame. These vectors are denoted by ρ. The point of interest is indicated by a following subscript. The reference frame of the vector is indicated by a preceding superscript. In other cases, the point of interest may be the center of gravity of the link or a load or a point of contact. In such cases, a different letter may be used to indicate the vector, but the subscript and superscript maintain their significance. The vector from the origin of frame A to the origin of frame B (or point of interest B) is ${}^{A}\rho_{B}$.

4.2.2 Rotations

4.2.2.1 Matrix representation

The rotation matrix is indicated by $\mathbb{R}$. The frame of origin is denoted by a preceding subscript while the target frame is indicated by a preceding superscript. The rotation matrix from frame B to frame A is of the form indicated in equation (4-1).

$$ {}^{A}_{B}\mathbb{R} = \begin{bmatrix} \hat{x}_B \bullet \hat{x}_A & \hat{y}_B \bullet \hat{x}_A & \hat{z}_B \bullet \hat{x}_A \\ \hat{x}_B \bullet \hat{y}_A & \hat{y}_B \bullet \hat{y}_A & \hat{z}_B \bullet \hat{y}_A \\ \hat{x}_B \bullet \hat{z}_A & \hat{y}_B \bullet \hat{z}_A & \hat{z}_B \bullet \hat{z}_A \end{bmatrix} \tag{4-1} $$

The rotation matrix gives no indication of location. The two frames of interest may be coincident or separated by any distance. Therefore, the rotation matrix from a given frame to any frames which are parallel are equal. That is, given frames C and D are parallel, the rotation matrices from frame E to C and E to D are equal.

$$ {}^{C}_{E}\mathbb{R} = {}^{D}_{E}\mathbb{R} \tag{4-2} $$

4.2.2.2 Euler angles

Euler angles express the orientation of one frame relative to another through three angle rotations. These rotations can be performed relative to fixed axes or to rotating axes. The three rotations can be converted to matrix form by expressing each of the rotations as a rotation matrix then multiplying them together. When rotating about fixed axes, each rotation should premultiply the rotations following it (chronologically). When rotating about rotated axes, each rotation should postmultiply the rotations following it

(chronologically). **Given rotations α, β, and γ, in that order**: about fixed axes, the composite rotation matrix is shown in equation 4-3,

$$\mathbb{R} = \mathbb{R}(\alpha)\mathbb{R}(\beta)\mathbb{R}(\gamma) \tag{4-3}$$

the composite rotation matrix for rotated axes is shown in equation 4-4.

$$\mathbb{R} = \mathbb{R}(\gamma)\mathbb{R}(\beta)\mathbb{R}(\alpha) \tag{4-4}$$

4.2.2.2.1 Hand pose

The hand pose is a vector representation of the position and orientation of a frame in space using Cartesian position coordinates and Euler angles. The hand pose, h, is a 6x1 vector. The first three elements are the position and the last three are the orientation in Euler angles as shown in equation 4-5.

$$h = \begin{bmatrix} x & y & z & \alpha & \beta & \gamma \end{bmatrix}^T \tag{4-5}$$

The hand pose representation requires an indication of the Euler rotations. Just two of the common rotations are XYZ-Fixed and ZYZ-Rotated. In the XYZ-Fixed rotation, a rotation is performed about the original *x*-axis, then the original *y*-axis, then the original *z*-axis. In Rotated rotations, each successive rotation is performed about the axes which resulted from the previous rotations, as discussed in Section 4.2.2.2. The order of rotations and whether the axes are fixed or rotated dictates the method for constructing the rotation matrix for the composite rotations as in equations 4-3 and 4-4, for example. Appendix B of Craig [2005] gives all 12 Euler angle sets and all 12 fixed angle sets.

4.2.3 Transformations

4.2.3.1 Rotation matrix and vector

To represent a vector in a new frame, e.g. vector ${}^{B}\rho_{P}$ in frame A, two transformations must be performed; a rotation to a frame parallel to the target frame, frame A, and a translation from the origin of frame B to frame A. An intermediary frame, frame B' is chosen which is parallel to frame A, permitting a simple translation between the two, and which is coincident with frame B, permitting a simple rotation between the two. In Figure 4-1, the rotation matrix ${}^{B'}_{B}\mathbb{R}$ rotates frame B to frame B'.

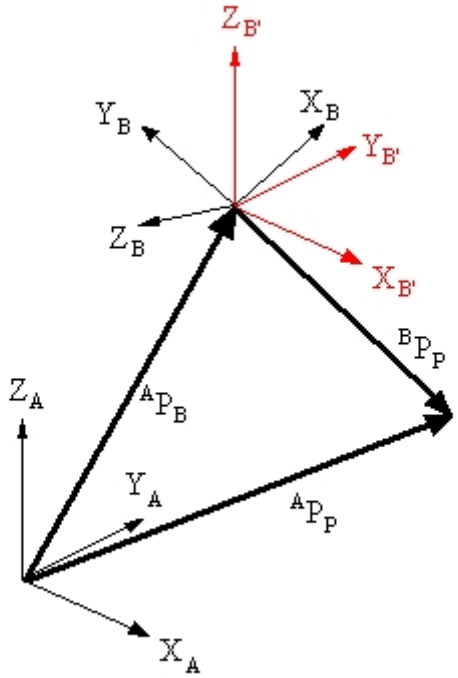


Figure 4-1: Example frame and vector transformation

Multiplication of the vector ${}^{B}\rho_{P}$ by the rotation matrix ${}^{B'}_{B}\mathbb{R}$ yields the location of the point P in frame B'.

$$ {}^{B'}\rho_{P} = {}^{B'}_{B}\mathbb{R}\,{}^{B}\rho_{P} \tag{4-6} $$

The translation between frame B' and frame A is now reduced to the addition of vector ${}^{A}\rho_{B'}$, which is equal to ${}^{A}\rho_{B}$. So, recognizing that ${}^{A}_{B}\mathbb{R}$ and ${}^{B'}_{B}\mathbb{R}$ are equal because A and B' are parallel, the complete representation of point P in frame A can be represented as

$$ {}^{A}\rho_{P} = {}^{A}_{B}\mathbb{R}\,{}^{B}\rho_{P} + {}^{A}\rho_{B} \tag{4-7} $$

Successive transformations can be carried out by performing the same calculation recursively.

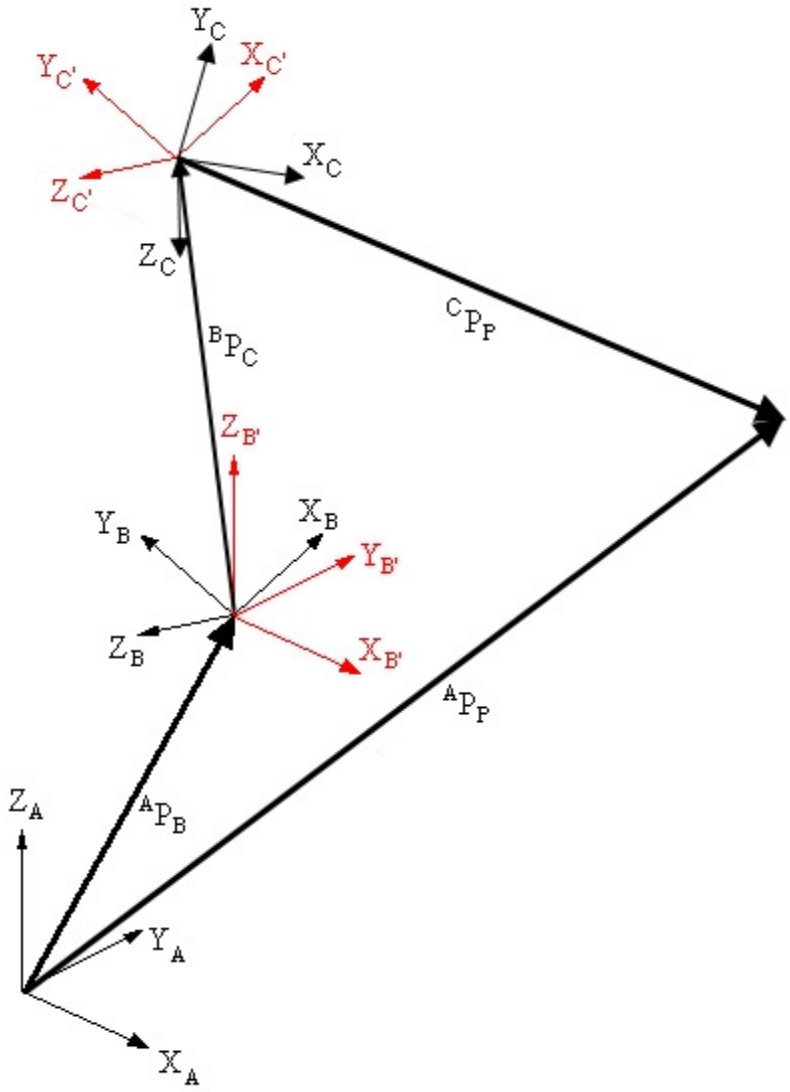


Figure 4-2: Example of successive transformations

Following equation 4-7 and given Figure 4-2, point P would be represented in frame B by ${}^{B}\rho_{P}$ in equation 4-8.

$$ {}^{B}\rho_{P} = {}^{B}_{C}\mathbb{R}\,{}^{C}\rho_{P} + {}^{B}\rho_{C} \tag{4-8} $$

Then, by substituting 4-8 into 4-7 it can be shown that the point P in frame A, ${}^{A}\rho_{P}$, is as in equation 4-9.

$$ {}^{A}\rho_{P} = {}^{A}_{B}\mathbb{R}\left({}^{B}_{C}\mathbb{R}\,{}^{C}\rho_{P} + {}^{B}\rho_{C} \right) + {}^{A}\rho_{B} \tag{4-9} $$

4.2.3.2 Transformation Matrices

Transformations can be expressed using a vector for the translation and a rotation matrix for the rotation. They can also be combined to facilitate consecutive rotation-translation pairs common to robotics. A 4x4 transformation matrix holds information about both the rotation and the transformation. The transformation matrix is constructed as in equation 4-10.

$$ {}^{A}_{B}T\{4\times4\} = \begin{bmatrix} {}^{A}_{B}\mathbb{R}\{3\times3\} & {}^{A}\rho_{B}\{3\times1\} \\ \underline{0}\{1\times3\} & 1 \end{bmatrix} \tag{4-10} $$

Using the 4x4 transformations, a translation following a rotation can be performed in one calculation. The vector to the point of interest needs only be premultiplied by the 4x4 transformation matrix. Because the matrix is 4x4, the vector must be 4x1. In order to make the math consistent, the 4th row of the new position vector will be unity. In this thesis, the 3x1 and 4x1 vectors representing position will be used interchangeably. So

$$ {}^{A}\rho_{B}\{4\times1\} = \begin{bmatrix} {}^{A}\rho_{B}\{3\times1\} \\ 1 \end{bmatrix} \tag{4-11} $$

In Figure 4-1 above, point P can be expressed in frame A by premultiplying the vector ${}^{B}\rho_{P}$ by the transformation ${}^{A}_{B}T$ as in 4-12.

$$ {}^{A}\rho_{P} = {}^{A}_{B}T\,{}^{B}\rho_{P} \tag{4-12} $$

Multiplying through, it can be seen that this is equivalent to the representation presented using a separate rotation matrix and translation vector. Substituting 4-10 into 4-12, it can be shown that the two representations are equivalent.

$$ {}^{A}\rho_{P} = {}^{A}_{B}T\,{}^{B}\rho_{P} = {}^{A}_{B}\mathbb{R}\,{}^{B}\rho_{P} + {}^{A}\rho_{B} \tag{4-13} $$

Successive transformations are performed by premultiplying by the next transformation going back to the target frame. For example, point Q, originally in frame C, represented in frame A is

$$ {}^{A}\rho_{Q} = {}^{A}_{B}T\,{}^{B}_{C}T\,{}^{C}\rho_{Q} \tag{4-14} $$

Returning to the example in Figure 4-2, point P in frame A would be calculated as shown in 4-15.

$$ {}^{A}\rho_{P} = {}^{A}_{B}T\,{}^{B}_{C}T\,{}^{C}\rho_{P} \tag{4-15} $$

4.2.4 Geometric representation summary

The representations above are useful in understanding Denavit-Hartenberg parameters and manipulator kinematics and dynamics. They will also be used in the algorithms presented in Chapter 5.

4.3 Denavit-Hartenberg Parameters

Development of kinematic and dynamic models requires a representation of the position and orientation of each manipulator link. Denavit and Hartenberg (D-H) [1955] developed a minimal set of parameters for describing the position and orientation of the joint axis of one rigid body relative to the next joint axis. Three of these parameters are fixed by the geometry of the manipulator and the last is dictated by the position of the actuator of the joint. Craig [2005] and Paul [1978] are the two commonly used notations for applying the D-H parameters to a serial manipulator. In this thesis, Craig's convention is used.

4.3.1 Frame assignment

An example of Craig's frame assignments is shown in Figure 4-3. In Craig's notation for the D-H parameters, the frame is assigned with the *z*-axis aligned with the rotational axis of the joint (or direction of displacement for a prismatic joint). The *x*-axis is parallel to the common normal between the axis of the current joint and the next. The common normal is the shortest distance between the two joint axes. The *y*-axis is determined using the right hand rule and the *x*- and *z*-axes. The origin of the frame is at the point where the common normal intersects the joint axis.

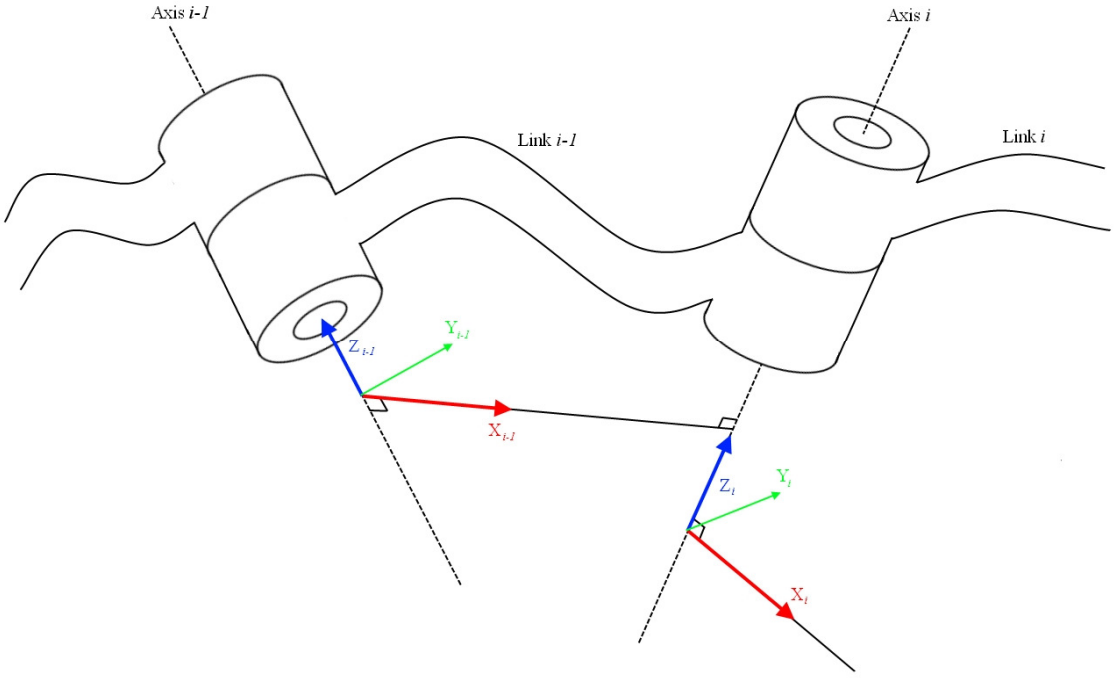


Figure 4-3: Denavit-Hartenberg frame assignment by Craig's notation

4.3.2 Parameter identification

The parameters are determined by the distances and rotations between two frames. Following the example in Figure 4-4, the first parameter is θ_i, the rotation between the common normal between axes *i-1* and *i* and the common normal between axes *i* and *i+1* as measured about axis *i*. For rotary joints, this is the parameter which changes with actuator position. The second parameter, d_i, is the distance between the common normals along axis *i*. For prismatic joints, this is the parameter which changes with actuator position. The distance along the common normal between axes *i* and *i+1* is a_i. Lastly, the angle between axes *i* and *i+1* as measured about the common normal between these axes is α_i.

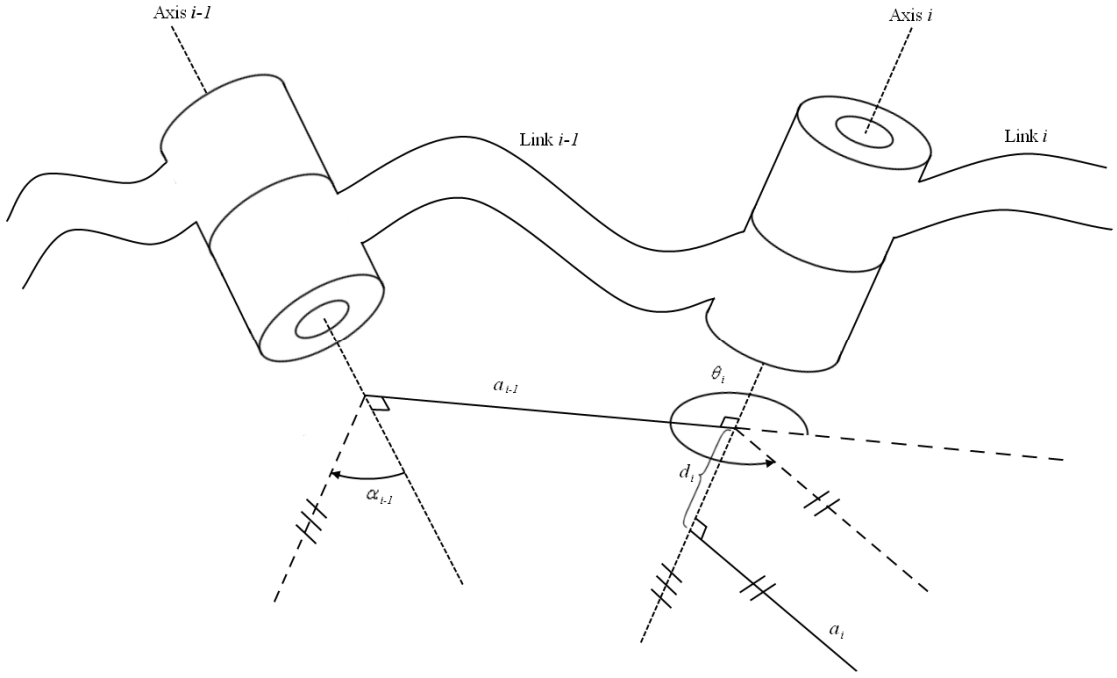


Figure 4-4: Denavit-Hartenberg parameters using Craig's notation

4.4 Force and velocity vectors

The force or velocity at a particular point on the manipulator is can be expressed in all 6 of their components using a free vector. Many properties of the force and velocity free vector are similar. Both of them express the translational and rotational components of the force and velocity properties as applied to a rigid body. Each is a function of the location on the rigid body and the reference frame whereas the fixed vectors used for position are functions only of the reference frame.

The key difference between the velocity and the force is the evaluation of the translational and rotational components at different points on the rigid body. The pure force, the translational force component, is consistent for a given reference frame over the

entirety of the rigid body. The force in frame i at point A, ${}^{A}F_i$ on the rigid body is the same as the force at point B on the same rigid body.

$$ {}^{A}\underline{F}_i = {}^{B}\underline{F}_i \tag{4-16} $$

The angular velocity, ω, is also the same at all points on the body. The rotation matrix is used to change the pure force (or angular velocity) from one frame to another.

$$ {}^{P}\underline{F}_A = {}^{A}_{B}\mathbb{R}\,{}^{P}\underline{F}_B \tag{4-17} $$

The 3-element pure force at point P as measured in frame A is represented by ${}^{A}F_P$ and ${}^{A}_{B}\mathbb{R}$ is the rotation matrix from frame B to frame A. Both the rotational component of force, the moment, and the translational component of the velocity must be calculated relative to their position on the rigid body. The moment at point A in frame A, ${}^{A}N_A$, is a function of the moment at B in frame B, ${}^{B}N_B$ as well as the force in frame B. [Craig 2005]

$$ {}^{A}\underline{N}_A = {}^{A}_{B}\mathbb{R}\,{}^{B}\underline{N}_B + {}^{A}\mathbf{P}_B \times {}^{A}_{B}\mathbb{R}\,{}^{B}\underline{F}_B \tag{4-18} $$

Where

$$ {}^{A}\mathbf{P}_B\times = \begin{bmatrix} 0 & -p_z & p_y \\ p_z & 0 & -p_x \\ -p_y & p_x & 0 \end{bmatrix} \tag{4-19} $$

If the general force and velocity vectors are represented as

$$\underline{F} = \begin{bmatrix} \underline{f} \\ \underline{n} \end{bmatrix} \tag{4-20}$$

$$\underline{V} = \begin{bmatrix} \underline{v} \\ \underline{\omega} \end{bmatrix} \tag{4-21}$$

where $\underline{f}$ is the 3-dimensional pure force, $\underline{n}$ is the 3-dimensional moment, $\underline{v}$ is the 3-dimensional linear velocity, and $\underline{\omega}$ is the 3-dimensional angular velocity, they must be transformed with two different transformation matrices. [Craig, 2005]

$${}^{A}_{B}\mathbf{T}_{f} = \begin{bmatrix} {}^{A}_{B}\mathbb{R} & 0 \\ {}^{A}\mathbf{P}_{B} \times {}^{A}_{B}\mathbb{R} & {}^{A}_{B}\mathbb{R} \end{bmatrix} \tag{4-22}$$

$${}^{A}_{B}\mathbf{T}_{v} = \begin{bmatrix} {}^{A}_{B}\mathbb{R} & {}^{A}\mathbf{P}_{B} \times {}^{A}_{B}\mathbb{R} \\ 0 & {}^{A}_{B}\mathbb{R} \end{bmatrix} \tag{4-23}$$

However, the velocity transformation matrix, ${}^{A}_{B}\mathbf{T}_{v}$, can be used for both the force and the velocity transformations if the representation of the force vector is defined as

$$\underline{F} = \begin{bmatrix} \underline{n} \\ \underline{f} \end{bmatrix} \tag{4-24}$$

In OSCAR, there exists a 6x6 transformation matrix object, the SpatialXform, which follows the format of the velocity transformation described above. No class exists specifically for six-element force vector representation. In order to preserve the use of the spatial transformation matrix already developed for OSCAR, the second formulation of force will be used.

The OSCAR Handpose representation of the position and orientation will be used to represent the 0th order kinematic state. Using these two representations makes control of the manipulator based on force inputs easier and more intuitive. The mathematical models for kinematics and dynamics will be used to demonstrate how force and torque data can improve the safety, reliability, and effectiveness of robotic manipulators in the next chapter.

4.5 Kinematics

It is necessary to determine the position and orientation of various points on the manipulator including tool points or extremities of the manipulator near an obstacle. When using torque sensing to detect collisions, the collision may happen at any link after the first joint. Determining the point of contact or the location of the point of interest requires a generalized model for the manipulator kinematics and dynamics which can be applied to a subset of the links of the manipulator. Generalized kinematics [Thomas and Tesar, 1982], dynamics [Freeman and Tesar, 1988], and compliance [Hernandez, 1996] models have been previously developed at the University of Texas at Austin. The kinematics model has been implemented in OSCAR [Kapoor, 1996]. Using the described transformations, the position and orientation, $\underline{u} \in R^m$, of any point attached to a manipulator can be determined for a given set of joint positions, $\underline{\phi}(t)$. The manipulator's DOF truncated at the point of interest is m.

$$u = f\left(\phi_1(t), \phi_2(t), \ldots, \phi_n(t)\right) \tag{4-25}$$

Taking the derivative with respect to time gives the output velocity as a function of the input velocities.

$$\underline{\dot{u}} = \frac{\partial \underline{u}}{\partial \underline{\phi}} \frac{\partial \underline{\phi}}{\partial t} = \left[G_{\phi}^{u} \right] \underline{\dot{\phi}} = \mathbf{J} \underline{\dot{\phi}} \tag{4-26}$$

The set of 1st-order influence coefficients which relates the change in input to change in output is the Jacobian, $\left[G_{\phi}^{u} \right]$ or $\mathbf{J}$. The Jacobian also relates joint accelerations to output accelerations. There are several different methods for determining the Jacobian values. [Thomas and Tesar, 1982] The Hessian accounts for Coriolis and centripetal effects arising from joint velocities. The overall output acceleration is therefore related to the joint velocities and accelerations.

$$\underline{\ddot{u}} = \frac{\partial \underline{u}}{\partial \underline{\phi}} \frac{\partial^2 \underline{\phi}}{\partial t^2} + \frac{\partial^2 \underline{u}}{\partial \underline{\phi}^2} \frac{\partial \underline{\phi}}{\partial t} = \left[G_{\phi}^{u} \right] \underline{\ddot{\phi}} + \underline{\dot{\phi}}^{T} \left[H_{\phi\phi}^{u} \right] \underline{\dot{\phi}} \tag{4-27}$$

4.6 Dynamics

4.6.1 Manipulator dynamics

A generalized torque model for a manipulator was developed by Freeman and Tesar. [1988] This model determines the actuator torques for given joint positions, velocities, and accelerations when under l loads, $L_{e,j}$. The vector of actuator torques, $\underline{\tau}_{\phi}^{total}$, is also a function of the inertia matrix $\left[I_{\phi\phi}^{*} \right]$, the power array $\left[P_{\phi\phi\phi}^{*} \right]$, the weight of each

link $L_{g,i}$, and the Jacobians calculated at the centers of gravity for each link $\left[{}^{i}G_{\phi}^{cg} \right]$ and at each point, e, where a load is applied $\left[{}^{j}G_{\phi}^{e} \right]$.

$$\underline{\tau}_{\phi}^{total} = \left[I_{\phi\phi}^{I} \right] \underline{\ddot{\phi}} + \underline{\dot{\phi}}^{T} \left[P_{\phi\phi\phi}^{*} \right] \underline{\dot{\phi}} + \sum_{i=1}^{n} \left[{}^{i}G_{\phi}^{cg} \right]^{T} \mathbf{L}_{g,i} + \sum_{j=1}^{n} \left[{}^{j}G_{\phi}^{e} \right]^{T} \mathbf{L}_{e,j} \tag{4-28}$$

The inertia and power matrices are functions of the joint positions and the physical properties of the links.

4.6.2 Torque due to contact force

In collision detection, the dynamic model is used to estimate the joint torques which are then compared to the measured values. The algorithm must be tuned for the manipulator to get accurate predicted values. There are several methods for determining the manipulator parameters as discussed in Chapter 2. Collision detection with joint torque sensors will be discussed in 5.2.3.2.

The measured joint torques can also be used to estimate the end effector forces by calculating the torques due to all the other dynamic loads: Coriolis and centripetal, gravity, and inertial.

$$\left(\left[G_{\phi}^{e} \right]^{T} \right)^{-1} \left[\underline{\tau}_{\phi}^{total} - \left[\mathrm{I}_{\phi\phi}^{*} \right] \underline{\ddot{\phi}} - \underline{\dot{\phi}}^{T} \left[P_{\phi\phi\phi}^{*} \right] \underline{\dot{\phi}} - \sum_{i=1}^{n} \left[{}^{i}G_{\phi}^{cg} \right] \underline{\mathbf{L}}_{g,i} \right] = \underline{\mathbf{L}}_{e} \tag{4-29}$$

This calculation assumes there is only one point of contact and that the point of contact is known ahead of time. The point of contact must be known in order to calculate the Jacobian at the point of contact, $\left[G_{\phi}^{e} \right]$.

4.6.3 Force transmission

When the manipulator has 6 DOF, the inverse of the transpose of the Jacobian at the point of contact can be multiplied by the resulting torque vector to get the full, six-axis force. When the joint configuration and end effector spaces are of a different dimension, a pseudo inverse must be used to calculate the end effector forces. The left-right pseudo inverse method was used for this calculation.

4.7 Chapter summary

A background on frame assignment and the representation of one frame relative to another is presented. The frames assigned on the manipulator are used to calculate the forward and inverse kinematics of the manipulator. Additionally, a dynamic model is presented. These tools will be necessary for the algorithms presented in the next chapter.

4.7.1 Notation summary

$^{A}\rho_{B}$ The origin of frame B expressed in frame A

$^{A}_{B}\mathbb{R}$ The rotation from frame B to frame A expressed as a 3x3 matrix

$^{A}_{B}T$ The 4x4 transformation used to transform 4x1 positional elements from frame B to frame A

$^{A}_{B}\mathbf{T}_{v}$ The 6x6 transformation used to transform 6x1 velocity and force (expressed as $F = \begin{bmatrix} \overline{n}^{T} & \overline{f}^{T} \end{bmatrix}^{T}$ vectors from frame B to frame A

τ_{total} Vector of total joint torques for all joints

τ_{int} Vector of joint torques due to link weights, Coriolis, and inertial effects

τ_{ext} Vector of joint torques due to contact forces

$\left[{}^{i}G_{\phi}^{cg} \right]$ Jacobian for joint *i* calculated to the center of gravity

$\left[{}^{j}G_{\phi}^{e} \right]$ Jacobian for load *j* calculated to the point of load application

5 Algorithms for Use of Force and Torque Data on Position Controlled Manipulators

5.1 Chapter Overview

Chapter 3 covered several force and torque modeling and control techniques and their research applications. Most of these have yet to be broadly adopted for commercial applications. More development and testing are necessary to build industry confidence and progress the body of industrial[2] knowledge. With more testing and applications, these more advanced algorithms may be deemed safe and effective; both by those making implementation decisions as well as those working directly with the robots in the glovebox. This chapter outlines force and torque data methods which are relatively simple to implement, sensor agnostic, and demonstrate the potential improvement in robot safety and manipulability.

5.2 Common Tools

Before presenting the algorithms, a few common concepts and calculations are presented. The terms used in this thesis to refer to different forces and torques are defined. Then the major frames of interest for these algorithms are presented. In order to gather the force and torque data, the sensors must be calibrated. Notes on sensor calibration are presented. The signal provided by the sensor contains noise and must be filtered to

[2] Here "industrial" refers to all uses outside the research lab including commercial and governmental such as manufacturing at the national labs and defense applications, etc.

minimize that noise. The method of filtering is presented here. Finally, methods for isolating the forces due to contact from the total measured force are presented.

5.2.1 Categorization of forces and torques

Forces and torques originate from sources both internal and external to the manipulator. Obvious internal sources are the links' inertias and the Coriolis effects. It is debatable whether the forces and torques due to the weights of the links are external or internal. While the mass is internal to the manipulator, gravitational effects are external. For the purposes of this thesis, they will be considered internal. Forces and torques due to contact with the environment are external sources. In summary, the internal sources of forces and torques are

- Link weights
- Link inertias
- Coriolis effects
- Friction – these effects are not modeled in this thesis

and the sole external source is

- Contact with the environment.

5.2.2 Frames of interest

There are several major frames of interest used in the algorithms described in this thesis.

- **Global Frame**

  Fixed frame at point of interest relative to the manipulator base. Often used as common frame to relate data from different sensors to the manipulator.

- **Manipulator Base Frame (and successive frames)**

  Base frame is used to relate manipulator positions to global frame. Successive frames are used to calculate manipulator kinematics and dynamics. These are labeled by the joint to which they correspond. The Manipulator Base Frame is also referred to as Joint 1 Frame.

- **End Effector Frame**

  Collective reference point for all fingers of hand or tool point of non-dexterous end effectors. Reference point at end of manipulator which has a fixed relationship to the last manipulator joint frame.

- **Tool/Fingertip Frames**

  Frames at the tip of a fixed tool or tips of the fingers of a dexterous manipulator. Kinematics of each individual finger relate position and orientation of the fingertip frame to the end effector frame.

- **Contact Frame(s)**

  Frame at the point of contact with the environment. Often coincident with the fingertip or end effector frame.

- **Workpiece Frame**

  Frame attached to the workpiece to give reference to other points on the workpiece.

- **Sensor Frame(s)**

  Sensor frames include vision sensor frames, which are related directly back to the

global frame, torque sensor frames, which are identical (coincident and parallel) to the manipulator joint frame at that point, and six-axis force sensors, which have a fixed relationship to the preceding manipulator joint frame. For wrist-mounted six-axis force sensors, the sensor frame will be fixed relative to the last joint frame (and end effector frame).

The methods of transformation discussed in Section 4.2 can be used to transform positional data from one frame to another. Velocity and force vectors can be transformed using the information presented in Section 4.4. When referring to a particular instance of one of these frames, the title will be capitalized (e.g. Global Frame). When referring to the frame in a general sense, it will not be capitalized (e.g. global frame).

5.2.3 Sensor Calibration

In order to get meaningful information from the sensors, the readings must be converted. The method of conversion is different for each sensor, particularly in the parameters of the equation of conversion. Techniques for calibrating six-axis force sensors and joint torque sensors are presented below.

5.2.3.1 Six-axis force sensor calibration

The output from the six-axis sensor will be a number of "counts" which must be converted to meaningful force and torque units, e.g. pounds and inch-pounds or Newtons and Newton-meters. The conversion factor should be provided by the manufacturer of the sensor. Verification (or determination) can be performed by applying known forces (or torques) in known directions and recording the counts and the forces. Plot the force on the

vertical axis and the counts on the horizontal. With a good, linear sensor, the data should form a line. The slope of this line is the force per counts. A sensor offset will be reflected in an offset at the vertical axis. It will be assumed throughout the rest of this thesis that the sensor readings are converted to accurate measurements of force and torque.

5.2.3.2 Joint torque sensor calibration

The joint torque sensors require a conversion from the number of counts to the joint torque. This conversion has two parameters, the torque offset, τ_{O}, and the torque gain, g_τ. The actual torque, τ, is related to the measured value, τ_{M} by the following equation.

$$\tau = g_\tau \tau_M + \tau_O \tag{5-1}$$

The process used to find these values very nearly follows the method presented by Ma et al. [1994].

The joint torque gain, g_τ, is found by comparing a known torque to the sensor measurement under the load of that torque for each joint. The measurement before applying the torque is indicated by ${}^0\tau_M$ and the measurement while applying the torque is indicated as ${}^1\tau_M$. Both have units of counts. An example is shown in Figure 5-1. The torque applied by the weight of the links is τ_G and the torque applied by the load is τ_L. Both of these can have any units of torque desired. These values are substituted into the equation describing torque in terms of measured torque for the manipulator without and with the load, respectively.

$$\tau_G = g_\tau \, {}^0\tau_M + \tau_O \tag{5-2}$$

$$\tau_G + \tau_L = g_\tau \, {}^1\tau_M + \tau_O \tag{5-3}$$

Subtracting the torque when under load from the torque before load and rearranging yields the following.

$$g_\tau = \frac{\tau_L}{\left({}^1\tau_M - {}^0\tau_M\right)} \tag{5-4}$$

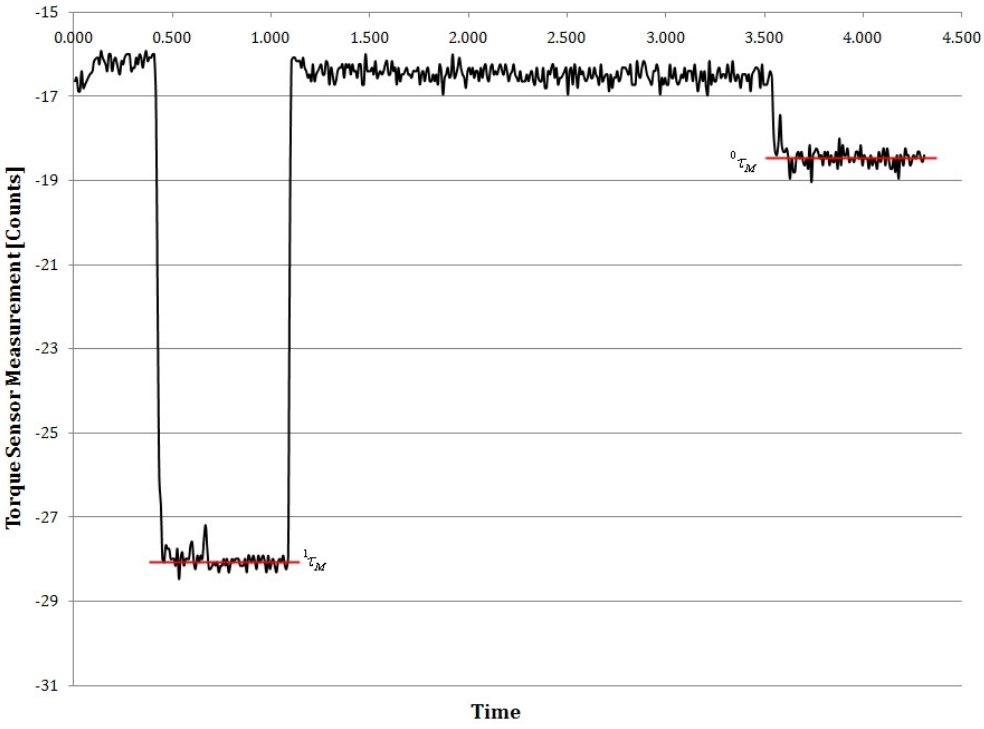


Figure 5-1: Example torque gain measurements

To find the value of the torque offset, each joint is moved through its full range while the rest of the joints are held stationary. The joint is moved slowly so that the torque due to

gravity is the only significant effect on the joint torque. When comparing the resulting joint position versus joint torque, the graph is sinusoidal. This sinusoid should be centered about zero because there exists on either side of a particular joint configuration a maximum torque due to gravity. The torque offset is the offset required to center the sine wave about 0. An example using the RRC K1207i is shown in Figure 5-2.

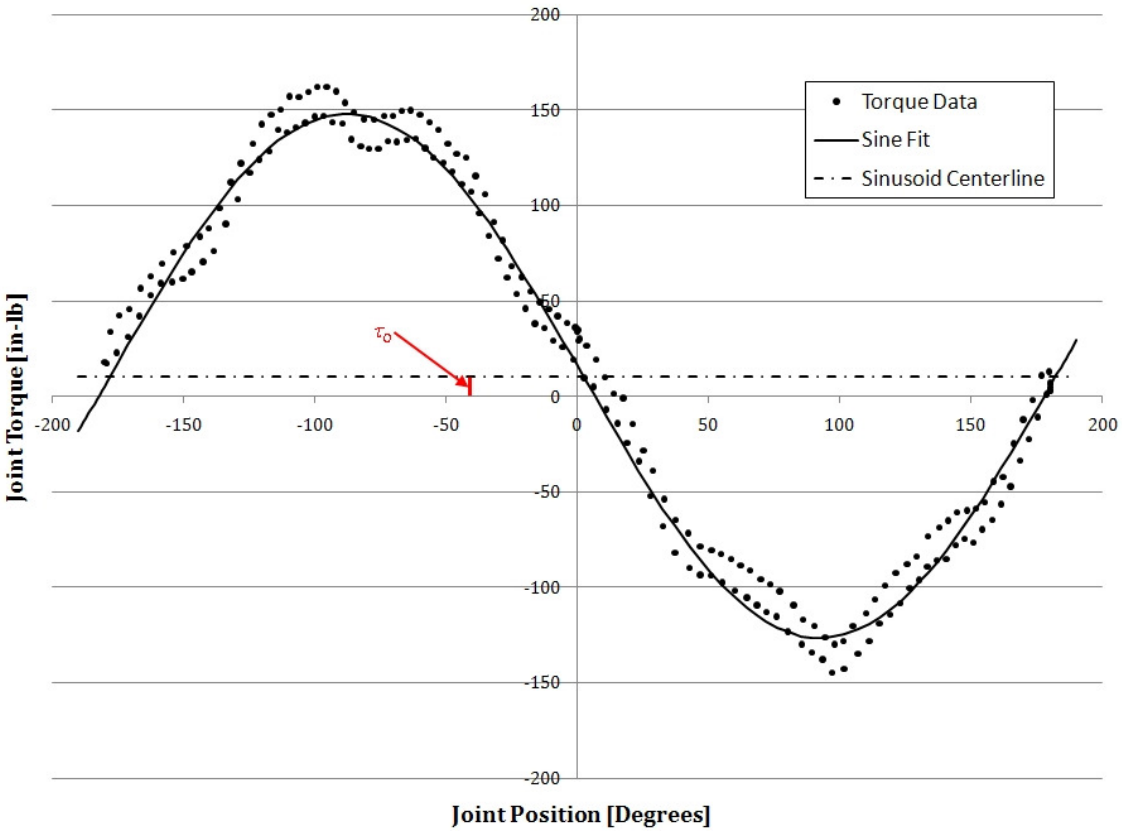


Figure 5-2: Example joint torque offset

5.2.4 Signal processing

One simple filtering technique is to average k data points before analysis. The analyzed data point, n_i, is equal to the sum of k measurements, m.

$$n_i = \frac{\sum_{j=(i-1)*k}^{i*k-1} m_j}{k} \tag{5-5}$$

Another simple filter for reducing sensor noise is a running average filter. Only the oldest element is replaced with the newest measurement. The data member analyzed, n_i, is equal to the sum of the previous k measurements, m, from the sensor.

$$n_i = \frac{\sum_{j=i}^{i+k} m_j}{k} \tag{5-6}$$

The filter smoothes some of the noise while still producing data at the same rate the sensor does. One disadvantage is that impulses may be averaged out, preventing a response.

5.2.5 External force isolation

Three force and torque sensors are commonly used in research and were reviewed in Section 2.5. Six-axis force sensors are nearly-rigid bodies which measure the forces and torques transmitted through the sensor. They measure the three forces (and three torques) along (and about) the principle axes. Joint torque sensors measure the torque about the rotary axis of each joint. Contact skins are mounted on the outside of links and measure force or sense pressure between the manipulator and the environment.

In safety and manipulability, the goal is to measure forces and torques due to contact and control the manipulator to manage these forces. Contact skins may be affixed to the end effector to measure contact forces directly. Because of the high cost and the

difficulty of affixing the skin without reducing the manipulator's dexterity, it is more common to use six-axis sensors and joint-torque sensors to measure the generated internal forces and torques.

At any point on the manipulator, the total forces and torques are the sum of the forces and torques due to internal and external sources.

$$F_{tot} = F_{ext} + F_{int} \tag{5-7}$$

The sensors discussed measure the total force and/or torque at a point. To react to contact forces, they must be isolated from the total force.

5.2.5.1 Six-axis force sensor – force isolation

Biasing the sensor subtracts the current force from all subsequent measurements. Now, if the forces due to internal sources are unchanged, all measurements will be a direct measure of the forces due to contact. Even if the magnitude remains the same, changing the direction of force at the point of sensing will change the readings from the sensor. Thus, to maintain a constant force due to internal sources, any change in orientation of the hand must be about the vertical axis only and all movements of the end effector must be quasi-static to minimize the forces and torques due to Coriolis and inertia.

It is not always possible to maintain constant forces and torques due to internal sources. If the manipulator's mass, center-of-mass, and inertia downstream of the sensor are known, the forces and torques due to internal sources can be calculated for any given

set of joint angles. Assuming quasi-static conditions, the end effector weight and forces due to contact are the only forces detected by the six-axis sensor.

5.2.5.1.1 Calculation of internal forces

Figure 5-3 shows a general end effector with a weight and a contact force relative to the sensor frame. In the figure, the sensor, contact, world, and center of gravity frames are denoted by the "S", "C", "W", and "cg" subscripts, respectively. For calculation of the weight, the force due to contact, F_C, must be zero.

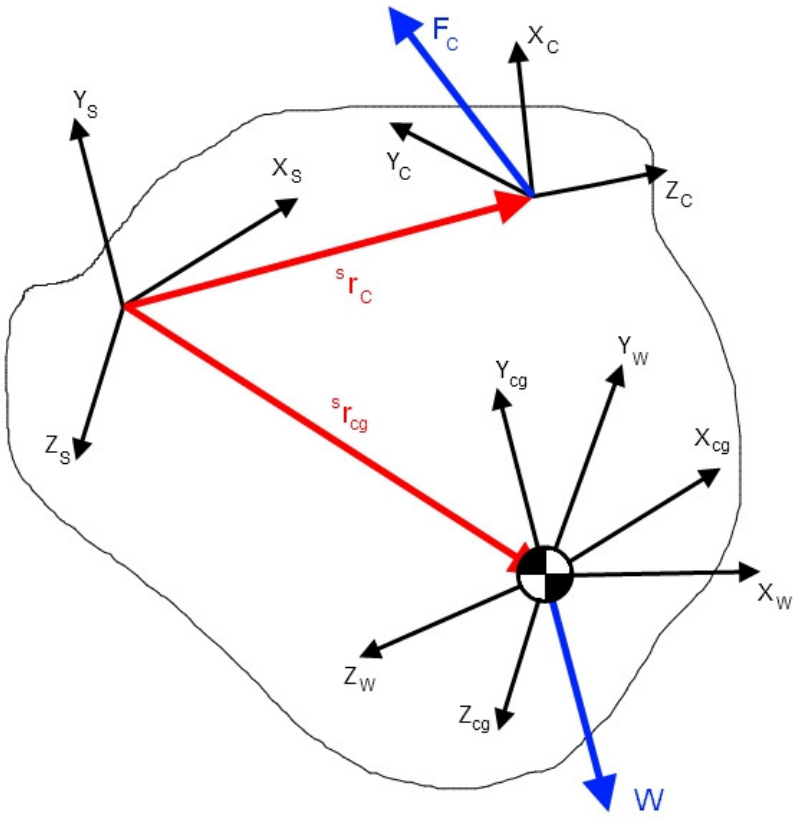


Figure 5-3: Generalized example of weight and contact force in local and sensor frames

To calculate the forces due to internal sources, i.e. the weight of the end effector, assume a frame, the center of gravity frame, which is parallel to the sensor frame with its origin at the center of gravity. Because it is parallel to the sensor frame, the rotations from any frame to the sensor frame or the center of gravity frame are equal. The weight of the end effector expressed in the center of gravity frame, ${}^{cg}W$, is

$$ {}^{cg}W = m\,{}^{S}_{W}\mathbb{R}\,{}^{W}\overline{g} \tag{5-8} $$

where m is the mass, ${}^{S}_{W}\mathbb{R}$ is the rotation from the world frame to the sensor (or CG) frame, and ${}^{W}\overline{g}$ is the gravity vector expressed in the world frame. The rotation matrix is dependent on the current manipulator configuration and can be calculated from forward kinematics.

There is no torque due to gravity, so the six-element force vector would be

$$ {}^{cg}\underline{F}_{int} = \begin{bmatrix} 0_{3x1} \\ {}^{cg}W \end{bmatrix}. \tag{5-9} $$

Using the velocity/force transform, the six-element force and torque vector expressed in the sensor frame, ${}^{S}\underline{F}_{int}$, is

$$ {}^{S}\underline{F}_{int} = {}^{S}_{cg}\mathbf{T}\,{}^{cg}\underline{F} = \begin{bmatrix} \mathrm{I}_{3x3} & \left[{}^{S}r_{cg}\times\right] \\ 0_{3x3} & \mathrm{I}_{3x3} \end{bmatrix} \begin{bmatrix} 0_{3x1} \\ m\,{}^{S}_{W}\mathbb{R}\,{}^{W}\overline{g} \end{bmatrix} \tag{5-10} $$

where $\left[{}^{S}r_{cg} \times \right]$ is the skew symmetric matrix for the vector from the origin of the sensor frame to the center of gravity as expressed in the sensor frame. (See equation 4-19 about the skew symmetric matrix.)

Equation 5-10 can be used to calculate the mass, m, and the vector to the center of mass in the sensor frame, ${}^{S}r_{cg}$. Knowing the mass and vector to the center of mass, equation 5-9 can be used to calculate the forces and torques due to internal sources. Subtracting this from the total (measured) force and torque at the sensor yields the force and torque due to external sources (contact).

5.2.5.1.2 Identification of point of contact

Equation 5-9 can also be used to calculate the force due to contact and the point of contact. The force at the sensor due to contact, measured in a contact frame parallel to the sensor frame, is

$$ {}^{S}\underline{F}_{ext} = {}^{S}_{cg}\mathbf{T}\,{}^{S}\underline{F}_{C} = \begin{bmatrix} \mathrm{I}_{3x3} & \left[{}^{S}r_{C} \times \right] \\ 0_{3x3} & \mathrm{I}_{3x3} \end{bmatrix} \begin{bmatrix} {}^{C}\underline{n}_{C} \\ {}^{C}\underline{f}_{C} \end{bmatrix} \tag{5-11} $$

where ${}^{S}\underline{F}_{C}$ is the six-axis contact force in the sensor frame, ${}^{S}r_{C}$ is the vector from the origin of the sensor frame to the origin of the contact frame expressed in the sensor frame, ${}^{C}\underline{n}_{C}$ is the torque due to contact about the contact frame, and ${}^{C}\underline{f}_{C}$ is the pure force at the point of contact expressed in the contact frame. In the general case, this yields 6 equations and 9 unknowns. Assuming a pure force at the point of contact (${}^{C}\underline{n}_{C} = 0$) or assuming the point

of contact is known reduces the set of unknowns to 6, making it possible to identify them all. The two component equations are shown.

$$ {}^{S}\underline{n}_{C} = {}^{S}_{C}\mathbb{R}\,{}^{C}\underline{n}_{C} + \left[{}^{S}r_{C} \times \right] {}^{S}_{C}\mathbb{R}\,{}^{C}\underline{f}_{C} \tag{5-12} $$

$$ {}^{S}\underline{f}_{C} = {}^{S}_{C}\mathbb{R}\,{}^{C}\underline{f}_{C} \tag{5-13} $$

5.2.5.2 Joint torque sensors – force isolation

For joint torque sensors, the force is not measured as a full six-axis force at one point but rather as torques about an axis at several locations on the manipulator. Referring back to equation 5-7, the total torque about the axis of measurement at each joint is the sum of the torque due to internal sources and the torque due to external sources.

$$ \tau_{total} = \tau_{int} + \tau_{ext} \tag{5-14} $$

As previously shown in equation 4-28, the torque due to internal sources is the sum of torques due to link weights and Coriolis and inertial effects.

$$ \underline{\tau}_{int} = \left[I^{I}_{\phi\phi} \right] \underline{\ddot{\phi}} + \underline{\dot{\phi}}^{T} \left[P^{*}_{\phi\phi\phi} \right] \underline{\dot{\phi}} + \sum_{i=1}^{n} \left[{}^{i}G^{cg}_{\phi} \right]^{T} \mathbf{L}_{g,i} \tag{4-28} $$

Assuming one point of contact, the torque due to external sources (the contact force), from equation 4-28, is expressed in equation 5-14.

$$ \underline{\tau}_{ext} = \left[G^{e}_{\phi} \right]^{T} \mathbf{L}_{e,j} \tag{5-15} $$

By using a model of the manipulator to predict the joint torque due to internal sources, the torque due to external sources is found using equation 4-29.

$$\underline{\mathbf{L}}_e = \left(\left[G_\phi^e \right]^T \right)^{-1} \left[\tau_{total} - \tau_{int} \right] \tag{4-29}$$

This contact force can be used for improving manipulator safety, the force can be managed to ensure effective manipulation of objects in the environment, and it can be used in conjunction with positional information to update the system world model.

5.3 Safety

Safe operation requires that forces are kept below a threshold to prevent damage to the manipulator and its surroundings. Using the procedure outlined above for identifying the forces due to contact, the manipulator safety can be improved by

- **Threshold monitoring** – The forces at the point of contact are monitored and kept below a threshold. This can be done in semi-autonomous or tele-operated modes.
- **Collision detection** – Anomalous forces arising from contact are detected.

The point of contact is not known prior to collision detection. Any anomalous force, at any point on the manipulator, is considered a collision.

5.3.1 Threshold monitoring

For threshold monitoring, the point of contact is assumed known. The force at the point of contact can be calculated. The method of determination is different for six-axis sensors and joint torque sensors. Both are discussed in Section 5.2.5.

The task of establishing thresholds is unique to every situation and merits little discussion here. Obvious limits may include manipulator payloads or limits due to workpiece frailty.

5.3.2 Collision detection

When moving in free space, the forces due to contact should remain zero. The techniques presented in Section 5.2.5 are used to determine the external forces. When this force is non-zero, a collision is detected. Since no model is perfect, there is a certain level of uncertainty that must be accepted before a collision is triggered. In the absence of this uncertainty buffer, the algorithm would be constantly indicating collisions as sensor noise and modeling deficiencies cause the measured value to differ from the predicted.

5.3.2.1 Model uncertainty

Due to the noise in the sensor signal and uncertainties which are certain to exist in the model, the measured forces will often differ from the modeled forces even when no collision is present. To account for the uncertainties, a third term, ${}^{S}\underline{F}_{uncertainty}$, is added to equation 5-7.

$$ {}^{S}\underline{F}_{total} = {}^{S}\underline{F}_{int} + {}^{S}\underline{F}_{ext} + {}^{S}\underline{F}_{uncertainty} \tag{5-16} $$

The measured forces due to uncertainty will usually vary between negative and positive bounds. A maximum uncertainty is chosen to account for the discrepancies between the model and the measured values.

The threshold may be chosen by doing a statistical analysis and selecting the threshold based on a desired confidence interval. However, because the uncertainty will vary depending on position, configuration, velocity, acceleration, unmodeled friction terms, and sensor noise, it would require a vast amount of data and analysis. In many cases, it is easier to choose the thresholds near the maximum error experienced during some moves and tune the thresholds depending on the desired sensitivity. If a collision detection algorithm which minimizes false positives is desired, the thresholds may even be chosen considerably higher than the maximum measured error.

5.3.2.2 Algorithm performance

Algorithm performance is different for six axis force sensors than for joint torque sensors. For either sensor, the force of collision cannot be determined exactly because the uncertainty force is bounded but unknown. Thus, the <u>minimum</u> force of contact is in the range

$$ {}^{S}\underline{F}_{total} - {}^{S}\underline{F}_{int} - {}^{S}\underline{F}_{threshold} < {}^{S}\underline{F}_{ext}^{\min} < {}^{S}\underline{F}_{total} - {}^{S}\underline{F}_{int} + {}^{S}\underline{F}_{threshold} \tag{5-17} $$

This can also be arranged to show

$$ 0 < \left| {}^{S}\underline{F}_{ext}^{\min} \right| < \left(2 * {}^{S}\underline{F}_{threshold} \right) \tag{5-18} $$

This is illustrated in Figure 5-4 below.

Both are subject to the same errors in force detection due to uncertainty but joint torque sensors are additionally dependent on the joint configuration (i.e. singularities, location, etc). Whereas a contact force (pure force) is transmitted directly to a six-axis force

sensor, the torques due to contact at each of the joints is dependent on the point of contact. A force which occurs near a joint is more difficult to detect at that joint than one which occurs far from the joint.

5.3.2.2.1 Six-axis force sensors – collision detection performance

With the six-axis sensor, the force due to contact only needs to be isolated from the forces due to internal sources before comparing the measured value to the predicted value, as in equation 5-15. The contact force is not known exactly, so the point of contact cannot be determined exactly.

5.3.2.2.2 Joint torque sensors – collision detection performance

As mentioned for collision detection in general, the measured torque must be outside the bounds of the predicted torque plus the threshold before any torque is attributed to collision. This is illustrated in the Figure 5-4 below.

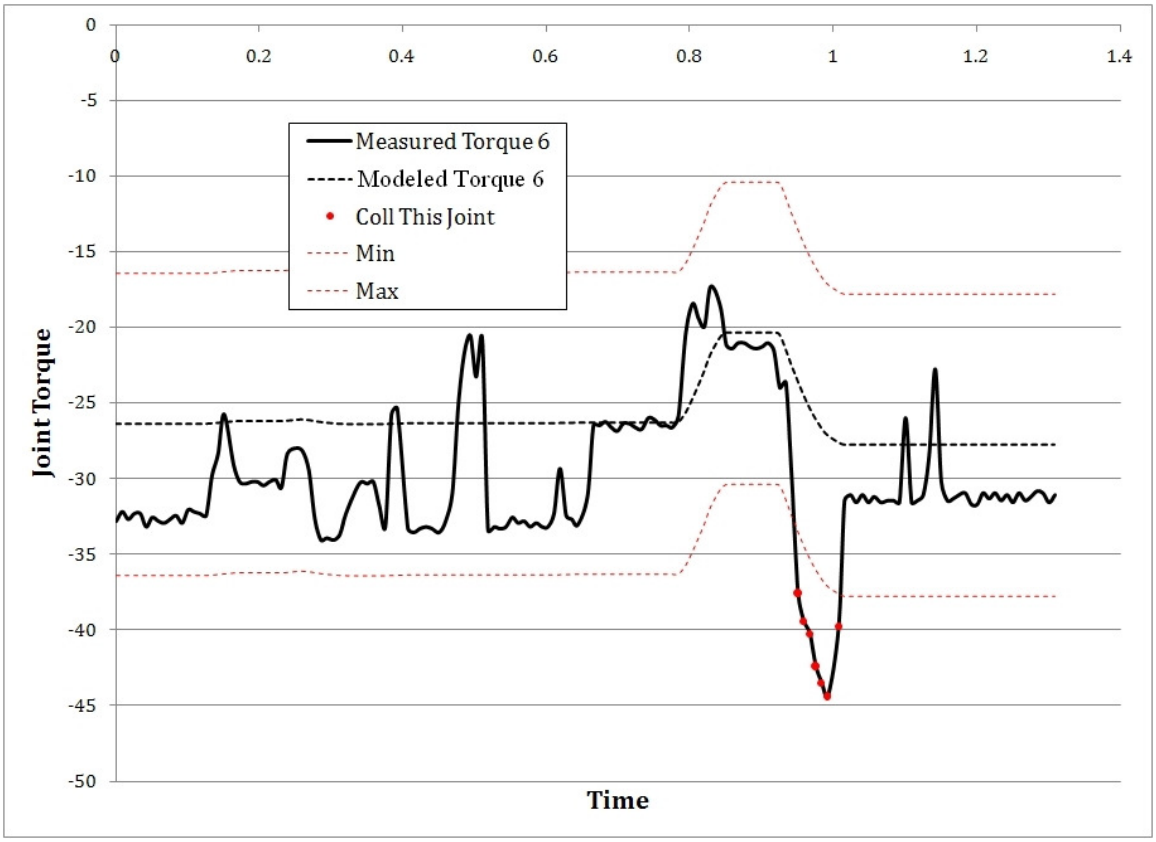


Figure 5-4: Example of collision detection using minimum and maximum threshold torques

If τ is non-zero, F must be non-zero unless the transpose of the Jacobian, $\left[G_{\varphi}^{e}\right]^{T}$, is singular. In case of a singularity, non-zero collision forces may not generate measurable torques at the joints. Even in a singular configuration, it is likely that the measurement at other joints would still reflect the collision except in certain extreme cases that are operationally rare. The collision's line-of-action would have to intersect or be parallel to all joint axes simultaneously. Such a collision would be detected by a six-axis force sensor but not by joint torque sensors. However, since the joint torques are not used to calculate force location, direction, or magnitude, this method should be sufficient for detection under virtually all normal operating conditions.

As shown in equation 5-17, the collision force may be nearly zero so the minimum force to trigger a collision is not useful. To determine the fidelity of the collision detection algorithm, it is more interesting to know the maximum contact force which might not trigger a collision. This is a complicated problem and is outside the scope of this thesis. More comments are made in Section 7.2.2.1.1.

5.3.3 Force limits

Without carefully chosen thresholds on the forces, there may be many falsely detected collisions or damaging, undetected collisions. The thresholds must be less than the safe limits for the manipulator and the environment but thresholds which are set too conservatively may excessively limit the manipulator's capabilities. Most industrial manipulators are composed of rugged (i.e. rigid) links connected by non-backdrivable joints. Thus very small changes in position directly after a collision can lead to large collision forces. Therefore, thresholds can be set relatively high if the operational bandwidth for reading the sensors is sufficiently high to detect these large forces *before* the continued motion of the manipulator continues along a potentially damaging path.

5.4 Managing contact forces using positional control

After monitoring the forces to ensure they remain below a threshold, the next logical step is to manage those forces. "Managing" means to operate the manipulator in an attempt to maintain the forces at a target value. The target value may be chosen to manipulate an object such as a door latch or it may be chosen to ensure contact between the

manipulator and the environment (a planned "collision"). The use of contact force to establish points of interests in the environment will be discussed in Section 5.5.

5.4.1 Force as input to position controller

During manipulation, reaction forces between the manipulator and environment must be managed (non-zero). Both their magnitude and direction are important in order to effectively manipulate an object. For example, a force in the wrong direction will not open a door and a force which is too little will not lift an object. The techniques discussed in Section 5.2.5 are used to determine the force due to contact depending on the sensor type.

Control of contact forces is well researched and has been discussed in Chapter 3. The purpose of this thesis is to demonstrate relatively simple and easy-to-implement algorithms. The demonstration presented here nearly follows the hybrid force/position control method presented by Craig [2005].

Craig presents natural and artificial constraints for each direction of force and position. A natural constraint is one in which the position or force is determined or limited by the physical configuration of the contact between objects. An example would be a block on a frictionless surface, as shown in Figure 5-5. In the -*z*-direction, position is constrained. After the block has made contact with the surface, it cannot continue to move in the negative *z*-direction. The forces parallel to the surface of the frictionless surface are constrained because the manipulator is not able to induce a force in either the *x* or *y* direction at the point of contact.

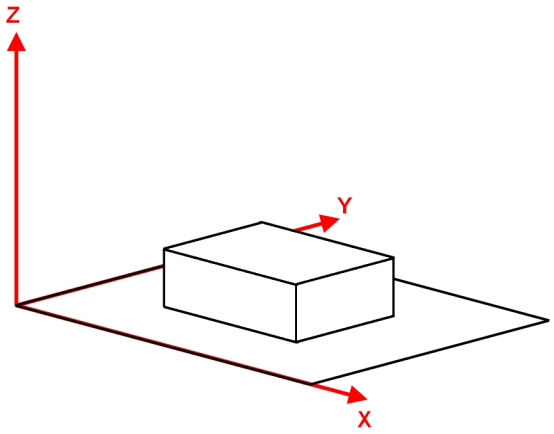


Figure 5-5: Position and force constraint example

In Craig's presentation, the manipulator appears to be force controlled. In this thesis the algorithms are developed for manipulators which are position controlled. Position constraints don't exist in the same manner described by Craig. Rather than being constrained, when the commanded end effector position goes beyond what Craig would consider a position constraint, the manipulator, end effector, and workpiece deform, generating forces. A very complicated model of the stiffness would be required to relate a particular displacement to the actual force. The stiffness would also be dependent on the particular manipulator configuration at the time. For the purposes of this thesis, it is only important that force is a function of the displacement.

$$F = f(\delta) \tag{5-19}$$

where F is the force and δ is the deformation. The relationship may not be linear, but a simplified example is shown in Figure 5-6 where the deformation occurs from point A to point B. As the deformation increases the force increases.

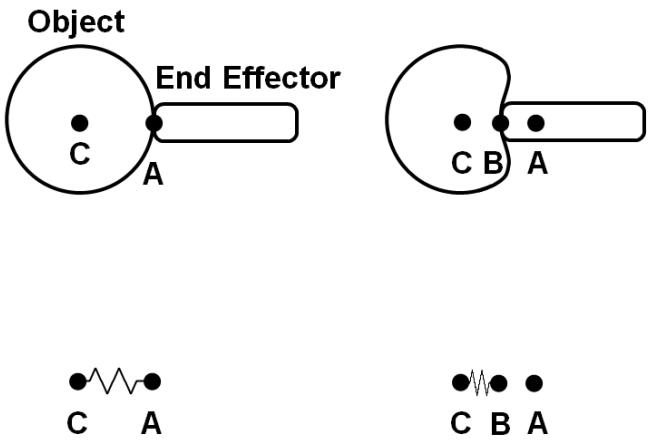


Figure 5-6: Object deformation spring model

The difference between the desired force and the measured force is the force error. When managing contact forces, the force error is used to control the position. The error is multiplied by a gain. This is used as a position error signal to change the manipulator position. The controller is shown in Figure 5-7 below. F_m is the measured force, q is the joint position, and F_d is the desired force.

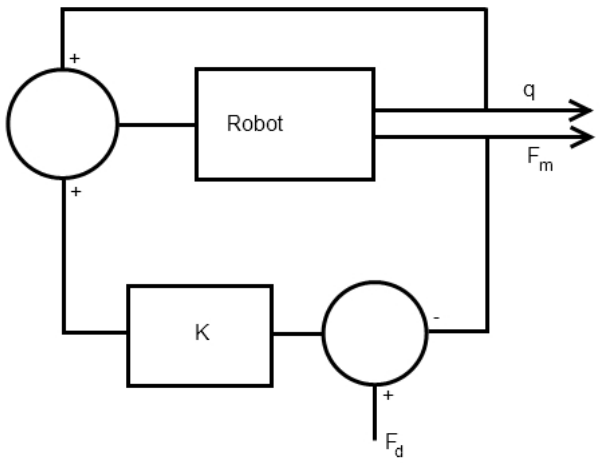


Figure 5-7: Controller Diagram

5.5 World Model Augmentation

In the previous section, the measured force is used to modify the end effector position, subsequently modifying the force. In this section, the force is used to identify the coincidence of the end effector with an object. First, a point is identified by noting the existence of non-zero contact forces. Next, the orientation of an object is identified by identifying several points simultaneously. Last, force data are used to identify features, such as curves, of an object in the environment.

5.5.1 Single points of contact

5.5.1.1 Identifying locations of points

Using the techniques mentioned above, the force is monitored to identify when a contact force exists. If the point of contact is also known, the location of that point can be identified in the global coordinate frame. In some cases, minimal knowledge about the end effector and the environment can be used to assume the point of contact. For example, when approaching a plane perpendicular with a pointed end effector, it can be assumed the tip of the end effector will come into contact first.

When using the six-axis force sensor and only one point is in contact, it is possible to use a combination of the forces and torques to determine the point of contact. The torque is related to the force by equation 5-19.

$$\tau = r \times F \tag{5-20}$$

where r is the position vector from the sensor to the point of contact in the sensor frame. Equation 5-19 can be solved for the position vector, resulting in equation 5-20.

$$r = \left[F \times\right]^T \tau \tag{5-21}$$

The matrix $\left[F \times\right]$ is the skew symmetric matrix for F and is defined as in equation 5-21.

$$\left[F \times\right] = \begin{bmatrix} 0 & -F_z & F_y \\ F_z & 0 & -F_x \\ -F_y & F_x & 0 \end{bmatrix} \tag{5-22}$$

So, given the measurements of force and torque, the point of contact can be determined relative to the sensor frame. Using the transformations presented in Section 4.2.3, the point of contact can be found relative to any other frame of significance, including the hand or global frame.

5.5.1.2 Identifying orientations of frames

In general, a frame can be defined using 3 non-collinear points. The frame defined is not unique to these points but it is useful to follow a specific procedure for defining these frames. The three points form a plane upon which two of the axes lie.

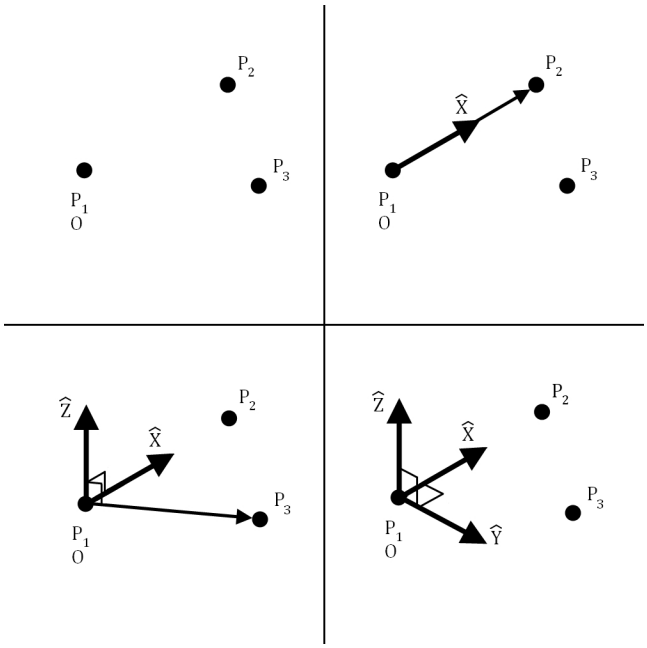


Figure 5-8: Frame identification using 3 points

Given the example in Figure 5-8, the first point, P_1, is the origin, O (eq 5-22). The unit vector from the first point to the second is the first axis, the *x*-axis (eq 5-23). The third point identifies the plane in which the second axis, the *y*-axis lies but the direction must be

determined after the *z*-axis is established. The cross product of the unit vector from the origin to the third point, P_3, and the *x*-axis is the *z*-axis (eq 5-24). The y-axis is finally determined as the cross product of the *z*- and *x*-axes (eq 5-25).

$$O = P_1 \tag{5-23}$$

$$\hat{X} = \frac{P_2 - P_1}{|P_2 - P_1|} \tag{5-24}$$

$$\hat{Z} = \left[\frac{P_3 - O}{|P_3 - O|}\right] \times \hat{X} \tag{5-25}$$

$$\hat{Y} = \hat{Z} \times \hat{X} \tag{5-26}$$

In some cases the frame may be modified as more data is collected. Take, for example, the identification of a target frame for the surface of a table, as illustrated in Figure 5-9. It will likely be easiest to first determine the plane in which two of the axes lie by identifying three points on the table top. Moving to one of the edges and establishing a vector along that edge gives a meaningful orientation to the three axes. When the corner or center of the table is found, the origin can be determined. In the absence of some of the data, these may be assigned arbitrarily.

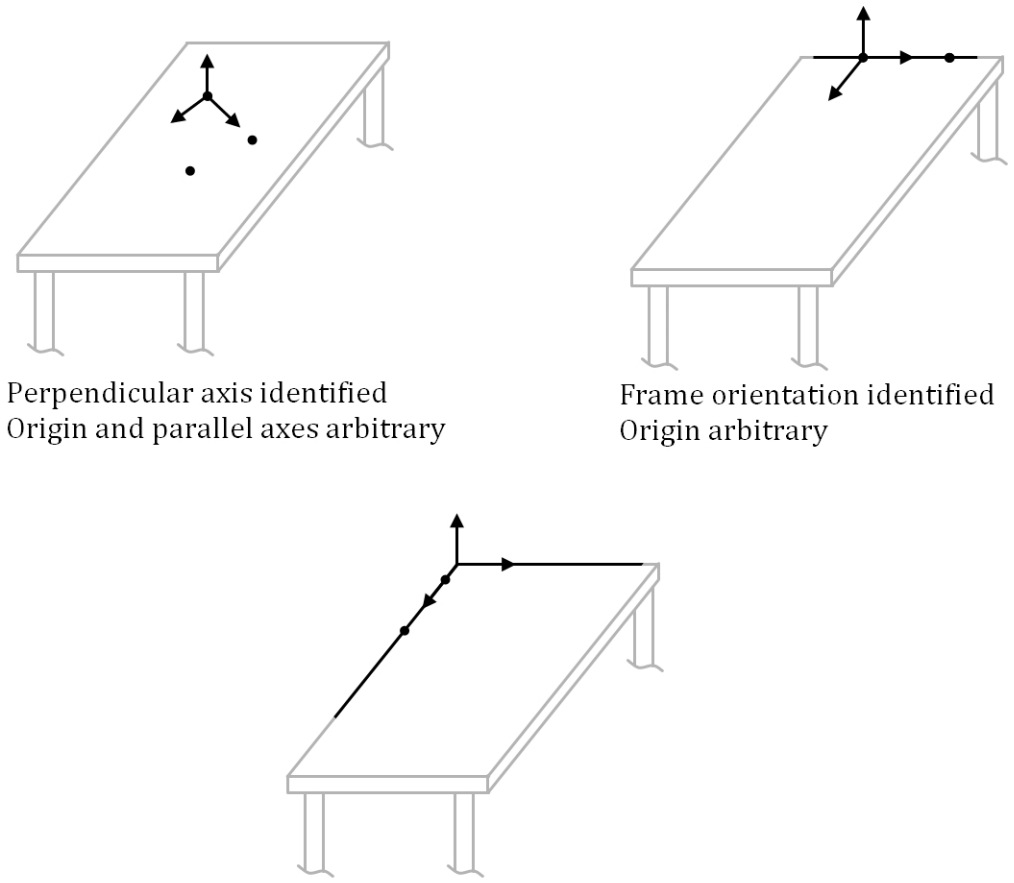


Figure 5-9: Modifying target frame assignment for table example

5.5.2 Multiple points of contact

Referring back to Figure 5-3, for multiple points of contact, the forces and torques due to external sources are the sum of the forces at each point of contact, C_i. For n points of contact, the total force due to external sources at the sensor would be expressed as in equation 5-26.

$$ {}^{S}\underline{F}_{ext} = \sum_{i=0}^{n} {}^{S}_{C_i}\mathbf{T}\,{}^{C_i}\underline{F}_i \qquad (5\text{-}27) $$

Given general forces and torques applied at each point and unknown points of contact, there are 9*n* unknowns; 3 positional data for each point of contact, 3 pure forces at the point of contact, and 3 pure moments at the point of contact. Solving equation 5-26 component-wise only gives 6 equations.

Visual data collected before beginning the algorithm allows the manipulator to approach the object given a few reasonable assumptions.

- **Pure forces at contact – moments are zero** – For simple geometries and well-planned approaches, the moment at the point of contact will be nearly zero. This reduces the number of unknowns to 6*n*.
- **Points of contact on end effector are known** – Given the visual data and proper algorithm planning, the object can be approached such that a limited set of end effector points first makes contact with the object. Knowing the points of contact reduces the unknowns to 3*n* – only the pure forces at the point of contact are unknown.
- **Geometry of object dictates force direction –** Using visual data to approach roughly perpendicular to the object, the dominant force will be perpendicular to the surface. Constraining the force directions gives 2*n* more equations for a total of 2*n*+3 and 3*n* unknowns.

There must be 3 points of contact (n=3) for there to be an equal number of equations and unknowns. Conveniently, 3 points are required to determine a plane or, as previously discussed, define a reference frame.

As discussed in Section 5.4.1, the force and displacement are related. Assuming the relationship between force and displacement are the same for each point of contact, equivalent forces come from equivalent displacements. To finely tune the orientation of the frame attached to the object, the forces should be balanced at each point of contact. If the relationship between force and displacement were exactly known, the balance would be

unnecessary; the displacement could be directly calculated from the force and the orientation of the object tuned from these data.

5.5.3 Identifying geometric features

Force data can also improve the world model by identifying positions while following a force-constrained path. A procedure like the one explained in Section 5.4.1 should be used to maintain a force while moving the manipulator. As the manipulator moves, the point of contact is recorded. A line or other curve of interest can be fit to the resulting data set to determine important environment parameters. For example, the contour of an object could be identified by moving the manipulator along the surface while maintaining a constant force.

5.6 Chapter Summary

This chapter discussed the techniques for using force data for safety, manipulability, and world-model augmentation. Sensed forces can identify a collision force and prompt a safe response. Potential reactions may include stopping or reversing the motion which led to the excessive force. A manipulator's ability to interact with its physical environment can be improved by using force and torque data as feedback. The world model can be improved by allowing the manipulator to identify the location, orientation, or contours of objects with which it establishes contact.

6 Algorithm Demonstrations

6.1 Chapter Overview

Two experiments were performed that demonstrate the algorithms proposed in Chapter 5.

- A six-axis force sensor mounted at the wrist of the 7-DOF Schunk LWA3 Lightweight Arm is used to open a cabinet door.
- Joint torque sensors on the Robotics Research Corporation (RRC) 7-DOF K1207i robotic arm are used to demonstrate collision detection

The experiments are detailed in the following sections.

6.2 Cabinet demonstration

In this demonstration, a cabinet door, shown in Figure 6-1, is opened using data with a relatively high level of uncertainty collected from a vision sensor and analyzed by the operator. The cabinet has a large, flat door that swings open about a hinge on the right side. The latch is operated by lifting. In this demonstration, all three areas of manipulator performance improvement presented in Sections 5.3, 5.4, and 5.5 are demonstrated using a six-axis force sensor.

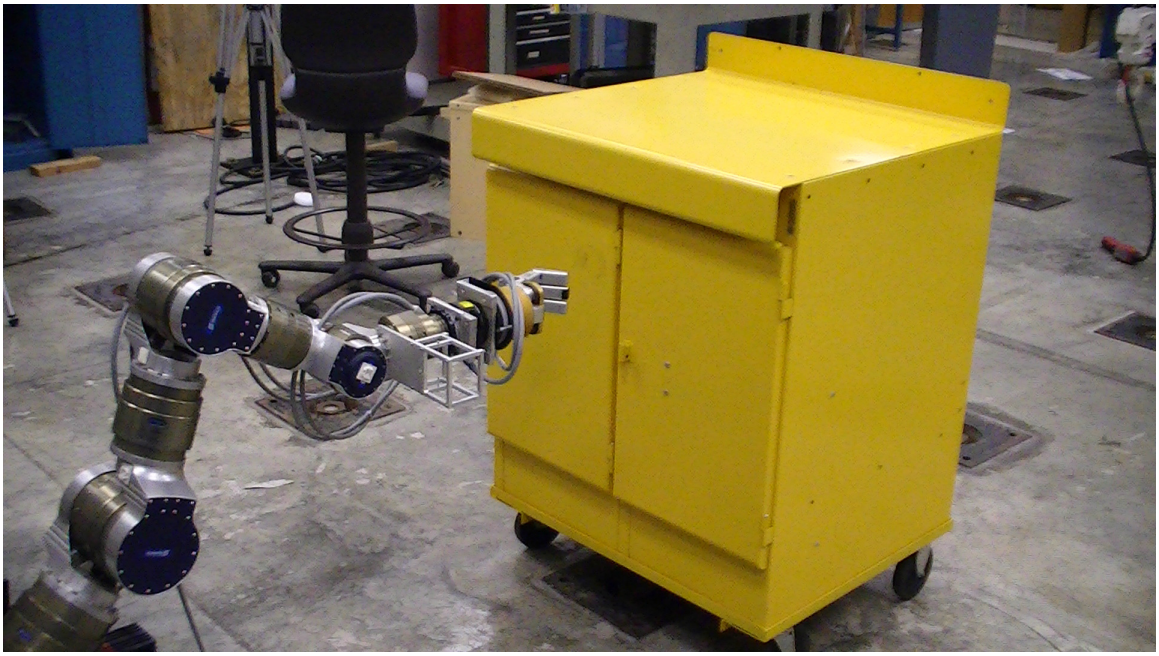


Figure 6-1: Manipulator and cabinet before opening demonstration

The algorithm makes a few operationally reasonable assumptions:

- The face of the cabinet is flat – used to fine tune the world model.
- The latch location can be roughly estimated by the operator from the visual data provided by the Swiss Ranger.
- The latch mechanism type can be identified and selected by an operator. In this case, the latch is opened by lifting a release mechanism on the bottom of the latch. In each case the latch type must be associated with a unique opening procedure.
- The door rotates on a fixed hinge which is parallel to the gravity vector so that the door neither swings open nor closed without assistance from the robot.
- Execution of the task is possible within the joint limits of the manipulator. i.e. the task is entirely within the manipulator's dexterous workspace. This assumption can be relaxed if the mobile platform is active during the task.

Note the assumptions are not designed to demonstrate a fully autonomous systems but rather demonstrate a realistically system that can more *safely* perform a given task and greatly reduce the burden on the operator.

6.2.1 System Setup

6.2.1.1 Manipulator and sensor types and models

A 7-DOF serial manipulator (the LWA3 Lightweight Arm) is mounted on a mobile manipulator (pictured in Figure 6-1). An ATI Gamma six-axis force sensor is mounted at the wrist, after the last joint. All data collected from the ATI Gamma six-axis force sensor is filtered through an averaging filter as described in Section 5.2.4. Two Swiss Ranger infrared 3-D Visual sensors are mounted near the base of the manipulator. These sensors provide a point cloud representation (albeit an uncertain one) of objects in their lines of sight. Because the challenges associated with integrating the data of two Swiss Rangers are not trivial and are outside the scope of this thesis, the cabinet is placed such that the task can be completed with the data from only one (but either) vision sensor. A Barrett Hand end effector is mounted at the end of the arm for manipulation.

6.2.1.2 Frame assignments

The base frame assignments for the cabinet opening demonstration are illustrated in Figure 6-2. The Global Frame origin is located at the top of the bottom plate of the mobile manipulator. The *y*-axis is parallel to the wheel axes. The *z*-axis points up such that gravity acts in the negative *z* direction. The *x*-axis points forward relative to the mobile platform. The manipulator is mounted such that the *z*-axis of the Manipulator Base Frame is +45° to

the Global *z*-axis as measured about the Global *y*-axis. The origin of the Manipulator Base Frame is offset from the Global Frame origin by roughly 470mm in the positive *x* direction and 420mm in the positive *z* direction resulting in a transformation of

$$ {}^{G}_{J1}T = \begin{bmatrix} & & & 470 \\ & \mathbb{R}(45, y) & & 0 \\ & & & 420 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{6-1} $$

The Swiss Ranger is mounted at (680mm, -235mm, 221mm) in the Global Frame and is rotated roughly +45° about the Global *z*-axis (before translation). The resulting transformation from Swiss Ranger coordinates to global coordinates is

$$ {}^{G}_{SW}T = \begin{bmatrix} & & & 680 \\ & \mathbb{R}(45, z) & & -235 \\ & & & 221 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{6-2} $$

Extensive calibration of this transform are unnecessary because the force sensor data will be used to compensate for small errors in position and orientation.

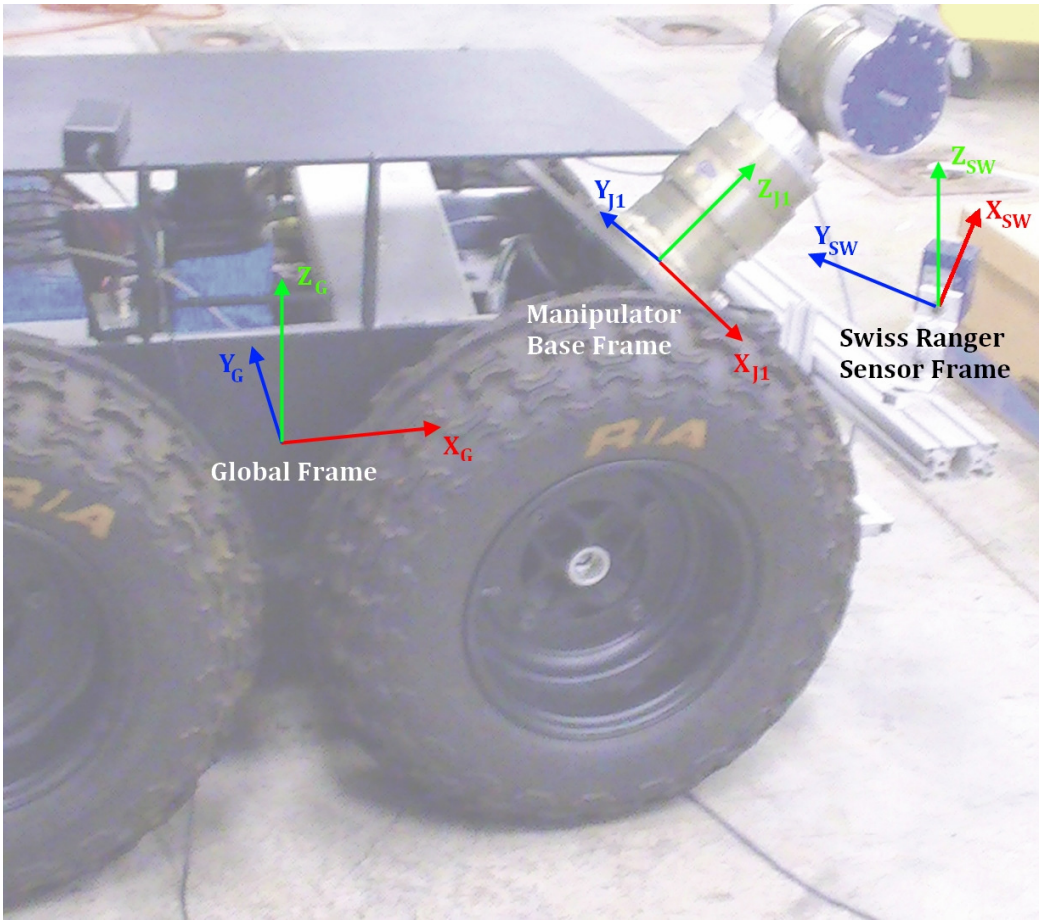


Figure 6-2: Cabinet demo base frame assignments

The ATI Gamma six-axis Force/Torque sensor is mounted at the end of the last joint of the manipulator as shown in Figure 6-3. The sensor has a fixed relationship to the last frame of the manipulator. The Sensor Frame is located at the center of the six-axis sensor and is oriented as marked by the manufacturer. The transformation from the Sensor Frame to the End Effector Frame is shown in equation 6-3. The Sensor Frame is shown relative to the last Manipulator Joint Frame and End Effector Frame in Figure 6-3.

$$ {}^{EEF}_{S}\mathbb{R} = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix} \tag{6-3} $$

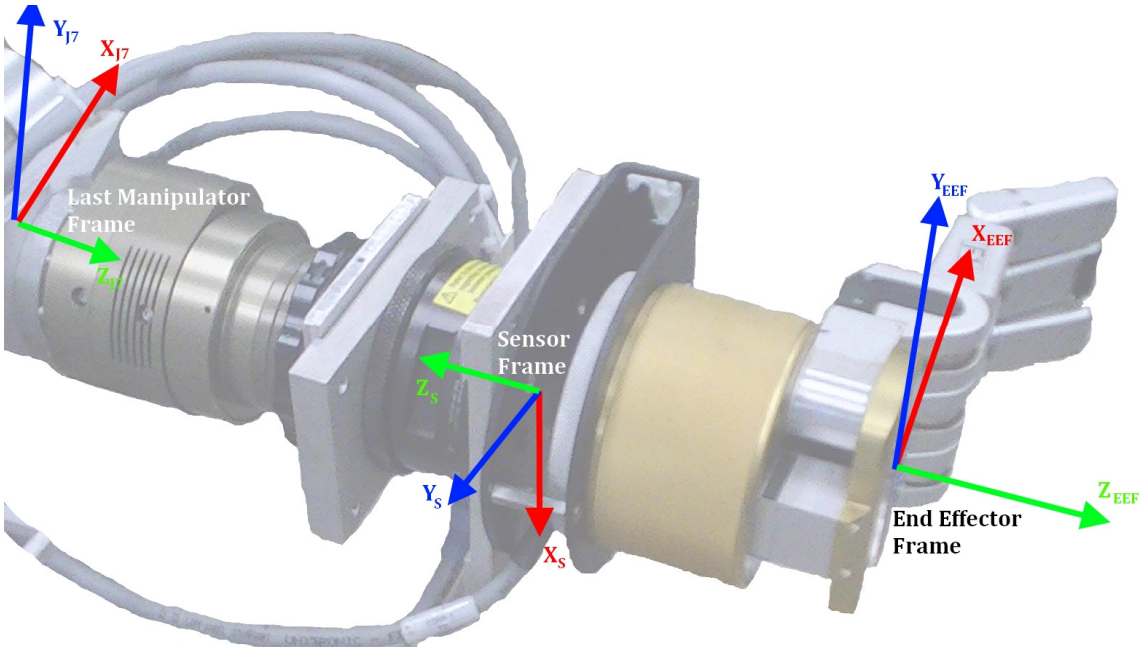


Figure 6-3: Sensor and End Effector Frame assignments

The Barrett Hand is attached to the end of the sensor as shown in Figure 6-3. All three of the Barrett Hand fingers have two "knuckle" joints which permit grasping. Each finger's knuckle motions are dependent on one another but each finger can grasp independent of the others. The Barrett Hand has two fingers which move opposite each other and one which is fixed relative to the hand. Thus the gripper has 4 controllable parameters: the spread of the fingers and the flexure of each of these three fingers. The End Effector Frame is attached to the Barrett Hand with the *x*-*y* plane parallel to the palm of the hand as shown in Figure 6-3 and Figure 6-4. The End Effector Frame is fixed relative to the

Sensor Frame. The *x*-axis points in the direction of the fixed finger. The origin of the End Effector Frame is located at the center of the palm of the Barrett Hand.

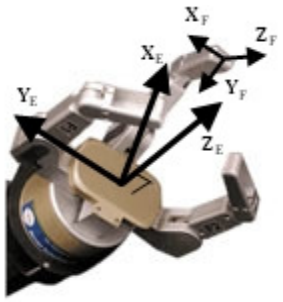


Figure 6-4: Barrett Hand with End Effector and Fingertip Frames

Frames are attached to the tips of each fingertip. The Fingertip Frames are related to the End Effector frame by the Barrett Hand forward- and inverse kinematics as described in the Barrett Hand manual.

6.2.2 Algorithm demonstration

6.2.2.1 Method summary

The method for opening the cabinet is broken into 5 steps.

1. Gathering and analyzing input data from Swiss Ranger and operator
2. Verify and/or refine the cabinet door location and orientation using contact force data
3. Lift latch
4. Pull door ajar
5. Push door open to approximately 90° from starting location

Each of the steps is detailed below. The use of force and torque data is emphasized.

With the exclusion of the first step, Gathering and analyzing input data, the forces and torques are monitored for safety (Section 5.3.1) during the entirety of the door opening process. If the forces due to contact exceed a threshold value unique to each step, the algorithm is stopped. For this demonstration, the End Effector or fingertip position and orientation are controlled in Global coordinates using Fixed XYZ rotations.

6.2.2.2 Gathering and analyzing input data

6.2.2.2.1 Cabinet door latch location

The cabinet door latch location is identified by the operator before execution of the algorithm begins. A GUI developed at UT Austin [Hulse, 2009] allows the operator to visualize the Swiss Ranger's 3D point cloud data. The operator clicks on the latch in the 2D visual image and the coordinates of the point in Swiss Ranger coordinates, ${}^{SW}\rho_{Latch}$, are automatically determined by mapping the selected pixel location to the 3D point cloud. The location of that point in the Global Frame is determined by transforming the measurement from the Swiss Ranger Frame to the Global Frame.

$$ {}^{G}\rho_{Latch} = {}^{G}_{SW}T\,{}^{SW}\rho_{Latch} \tag{6-4} $$

This point is used as the origin of a Workpiece Frame.

6.2.2.2.2 Cabinet door face orientation

After the latch location has been identified by the operator, an estimation of the orientation of the door is made. The Workpiece Frame is oriented using techniques similar to those discussed in Section 5.5.1.2 except the points are identified using vision sensor data

instead of force data. Several points each side of the latch are selected from the point cloud. 20 equally-spaced points to the left and right of the latch are extracted from the point cloud (Figure 6-5). These points are transformed to the Global Frame and a line is fitted to these points. This is the first vector for the Workpiece Frame and indicates the direction of the Workpiece Frame *x*-axis. The Workpiece Frame *z*-axis is determined directly by placing a unit vector parallel to the Global Frame *z*-axis with its origin at the Workpiece Frame origin.

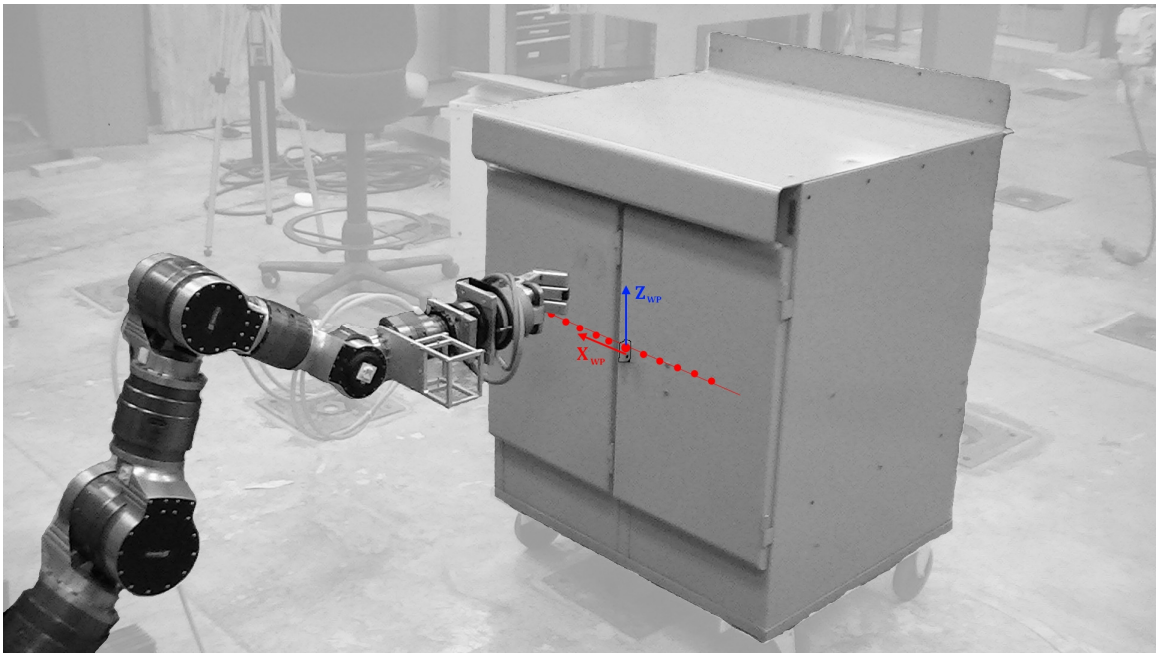


Figure 6-5: Establishing Workpiece Frame from vision sensor data

This frame will be used as the initial Workpiece Frame. Ideally, the Workpiece Frame *x*-axis is parallel to the Global *x*-*y* (horizontal) plane and along the face of the cabinet. The origin of the Workpiece Frame is at the latch as identified by the operator. In reality, there are errors due to tolerances in the Swiss Ranger point cloud data. The manipulator will approach the cabinet a few centimeters below the origin of the Workpiece Frame and with

the extended fingers initially oriented to form a plane parallel to the *x*-*z* plane (cabinet door) of this Workpiece Frame.

6.2.2.3 Verify cabinet door location and orientation

In this step, force and torque data will be used to verify the location and orientation of the face of the cabinet as estimated by the vision sensor data in the previous step. The manipulator will tactilely identify the distance to the face of the door and the orientation of the door. The principles from the following sections will be used:

- 5.3.1 – forces are monitored and the algorithm reacts if a threshold is exceeded
- 5.5.1 – verifying distance to cabinet face
- 5.5.2 – verifying cabinet face orientation

Throughout this step, the forces and torques are monitored to ensure they stay below the defined safe threshold. In this demonstration, if the force exceeds 4 lbs, the manipulator is stopped and the operator is notified of the situation. The threshold values were chosen below the operational capacity of the Barrett Hand. The operator takes any necessary action before restarting the algorithm.

The Barrett Hand is oriented to facilitate collection of force data for identification of frame orientation as described in Section 5.5.2. The fingers are spread such that they are separated by 120° about the End Effector Frame *z*-axis as shown in Figure 6-6. The fingers are extended so the distance along the End Effector Frame's *z*-axis from the origin of that

frame to the origin of the Fingertip Frame is several centimeters. This ensures the fingertips make contact with the cabinet face before the latch hits the palm of the hand.

The hand is moved to a position 10 mm in the negative Workpiece *z* direction and 50 mm in the negative Workpiece *y* direction from the Workpiece Frame (operator-identified latch location).

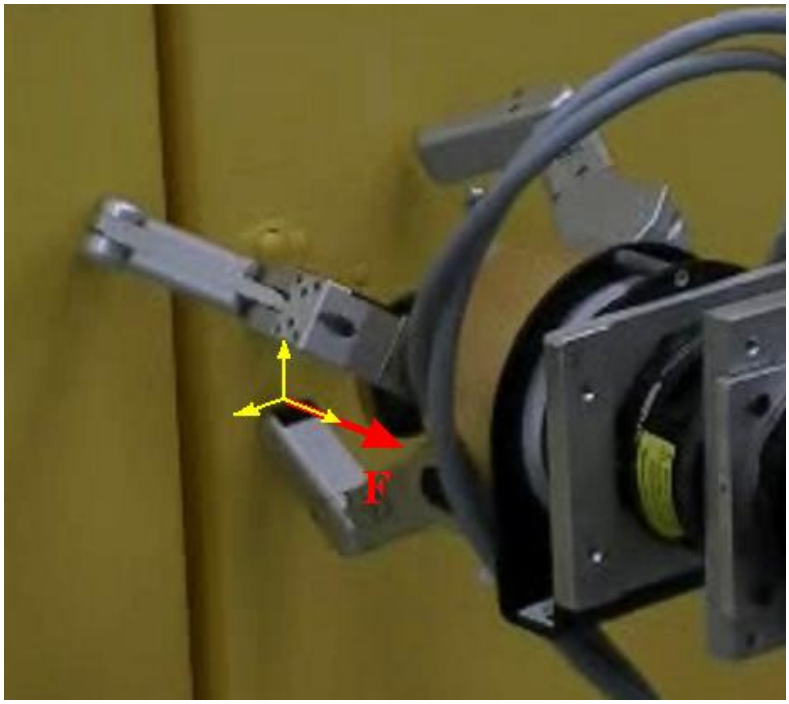


Figure 6-6: Establishing contact with the door

After being moved to this position, the six-axis sensor is biased. From this point through the end of the fourth step: pulling the door ajar, the angle between the gravity vector (Global *z*-axis) and the End Effector *z*-axis does not change so the forces due to the weight of the End Effector do not change. As a result, the measurements taken by the force

sensor exclude end effector weight, etc. The measurement still has some noise but the filter helps to reduce this.

The hand is then moved in increments of 1mm in the positive End Effector Frame *z* direction until a contact force (red vector labeled "F") is detected, as in Figure 6-6.

The Workpiece *z*-axis was identified as parallel to the Global *z*-axis because the cabinet face was assumed vertical. To confirm the orientation of the Workpiece frame aligns with the cabinet face, only the Workpiece *x*-axis needs to be verified. To do that, the techniques from Section 5.5.2 are used to identify two points on the face of the cabinet simultaneously.

Equation 5-26 relates the measured forces and torques to the forces at the points of contact.

$$ {}^{S}\underline{F}_{ext} = {}^{S}_{F1}\mathbf{T}^{F1}\underline{F} + {}^{S}_{F2}\mathbf{T}^{F2}\underline{F} + {}^{S}_{F3}\mathbf{T}^{F3}\underline{F} \tag{6-5} $$

Expanding this equation (showing only terms for contact force at first fingertip – full equation includes all three fingers):

$$ {}^{S}\begin{bmatrix} n \\ f \end{bmatrix}_{ext} = \begin{bmatrix} {}^{S}_{F1}\mathbb{R} & \left[{}^{S}r_{F1} \times \right] {}^{S}_{F1}\mathbb{R} \\ 0 & {}^{S}_{F1}\mathbb{R} \end{bmatrix}^{F1}\begin{bmatrix} n \\ f \end{bmatrix}_{C1} + \cdots \tag{6-6} $$

Because the door is flat and the fingertips are rounded with large radii, the torques at the points of contact are zero.

$$ {}^{S}\begin{bmatrix} n \\ f \end{bmatrix}_{ext} = \begin{bmatrix} {}^{S}_{F1}\mathbb{R} & \left[{}^{S}r_{F1} \times \right] {}^{S}_{F1}\mathbb{R} \\ 0 & {}^{S}_{F1}\mathbb{R} \end{bmatrix} {}^{F1}\begin{bmatrix} 0 \\ f \end{bmatrix}_{C1} + \cdots \tag{6-7} $$

Evaluating for only the torque at the sensor due to external sources yields

$$ {}^{S}n_{ext} = \left[{}^{S}r_{F1} \times \right] {}^{S}_{F1}\mathbb{R}\, {}^{F1}f_{C1} + \left[{}^{S}r_{F2} \times \right] {}^{S}_{F2}\mathbb{R}\, {}^{F2}f_{C2} + \left[{}^{S}r_{F3} \times \right] {}^{S}_{F3}\mathbb{R}\, {}^{F3}f_{C3} \tag{6-8} $$

Expanding, applying transformations, and isolating the torque about the Sensor *y*-axis (Global *z*-axis):

$$ {}^{S}n_{ext,y} = {}^{S}r_{F1,x}\, {}^{S}f_{C1,z} - {}^{S}r_{F1,z}\, {}^{S}f_{C1,x} + \cdots \tag{6-9} $$

Before approaching the door, the fingertips were oriented such that they would approach the cabinet perpendicular to the estimated face of the door. The contact force between each finger and the door is assumed to be normal to the face (no frictional component). Thus the force is primarily in the direction of the Sensor Frame's *z*-axis. Because the tip of the fixed finger, finger 3, of the Hand is neither left nor right of the origin of the Sensor Frame as described above and shown in Figure 6-6, ${}^{S}r_{F3,x}$ is 0 and equation 6-9 reduces to:

$$ {}^{S}n_{ext,y} = {}^{S}r_{F1,x}\, {}^{F1}f_{C,x} + {}^{S}r_{F2,x}\, {}^{F2}f_{C,x} \tag{6-10} $$

The other two fingertips are equally distant to the left and right of the Sensor origin such that ${}^{S}r_{F1,x} = -{}^{S}r_{F2,x}$, so

$$ {}^{S}n_{ext,y} = {}^{S}r_{F1,x}\left({}^{S}f_{C1,z} - {}^{S}f_{C2,z} \right) \tag{6-11} $$

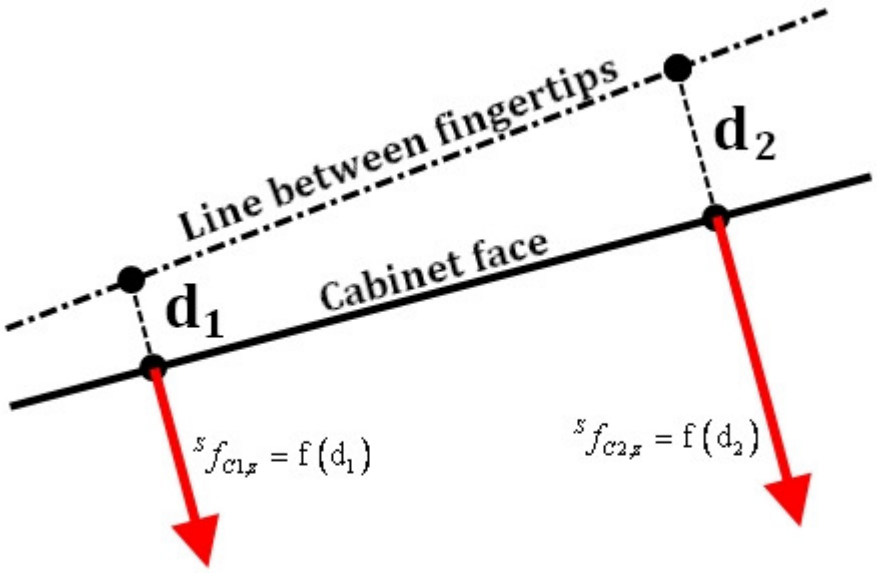


Figure 6-7: Contact between fingertips and cabinet door for alignment with face

Figure 6-7 illustrates the contact forces generated from contact between the fingertips and the cabinet door. The force, ${}^{S}f_{C1,z}$ or ${}^{S}f_{C2,z}$, at each point of contact is a function of the respective displacement due to contact, d_1 or d_2. The forces must be non-zero to ensure contact with the door. They must be equal to indicate that the displacement between the commanded fingertip position and the initial point of contact on the door is equal. The displacements must be equal so that the line between them is parallel to the face of the cabinet. Being parallel to the face of the cabinet, the fingertip positions can be used to identify the plane describing the face. This plane is used, as described in Section 5.5.1.2, to reorient the Workpiece Frame.

According to equation 6-11, the torque about the sensor *y* axis due to external sources must be zero to verify the orientation of the cabinet face. The angle about the point of contact, as measured about the Global *z*-axis, is modified until the torque is zero. The change in angle about the Global *z*-axis, $\Delta\gamma$, is a function of the measured torque;

$$\Delta\gamma = -K\,{}^{S}n_{ext,y} \tag{6-12}$$

where K is some positive gain. The gain is 0.05 degrees per inch-pound.

In this step, the force and torque data are used to verify the orientation of the Workpiece frame, similar to the second step of the example illustrated in Figure 5-9: Modifying target frame assignment for table example. In the next step, the latch will be lifted.

6.2.2.4 Lift latch

In this step, the end effector moves up the face of the door until contact is made with the latch. After making contact, the latch actuation/lifting force is managed in order to effectively unlatch the door. During this step, the following principles from Chapter 5 are used.

- 5.3.1 "Threshold monitoring" – forces are monitored and the algorithm reacts if a threshold is exceeded
- 5.4.1 "Force as input to position controller" – maintaining contact with the door face and lifting the latch

- 5.5.1.1 “Identifying locations of points” – identifying location of the latch

Throughout this step, the forces are monitored for safety. If any of the forces exceed 4 pounds, the manipulator stops and the operator is notified of the excessive force.

A contact force perpendicular to the face of the cabinet is maintained as the gripper comes into contact with the latch. This force is controlled via a positional controller to 0.7 pounds – sufficient to release the locking mechanism. The cabinet face is assumed vertical for orientation of the Workpiece Frame, but by maintaining a small perpendicular contact force, the gripper’s finger follows the face even if it is slightly out of the vertical plane.

The contact force is kept low in order to minimize the frictional forces, which point downward. The latch-lifting force, as seen by the sensor, will also point downward. If the frictional forces are too large, the force due to contact with the latch will not be distinguishable from the friction force or the constantly-monitored threshold force will be exceeded. This is illustrated in Figure 6-8 with the frictional force (blue arrow) pointing down and the contact normal force (red arrow) pointing toward the sensor.

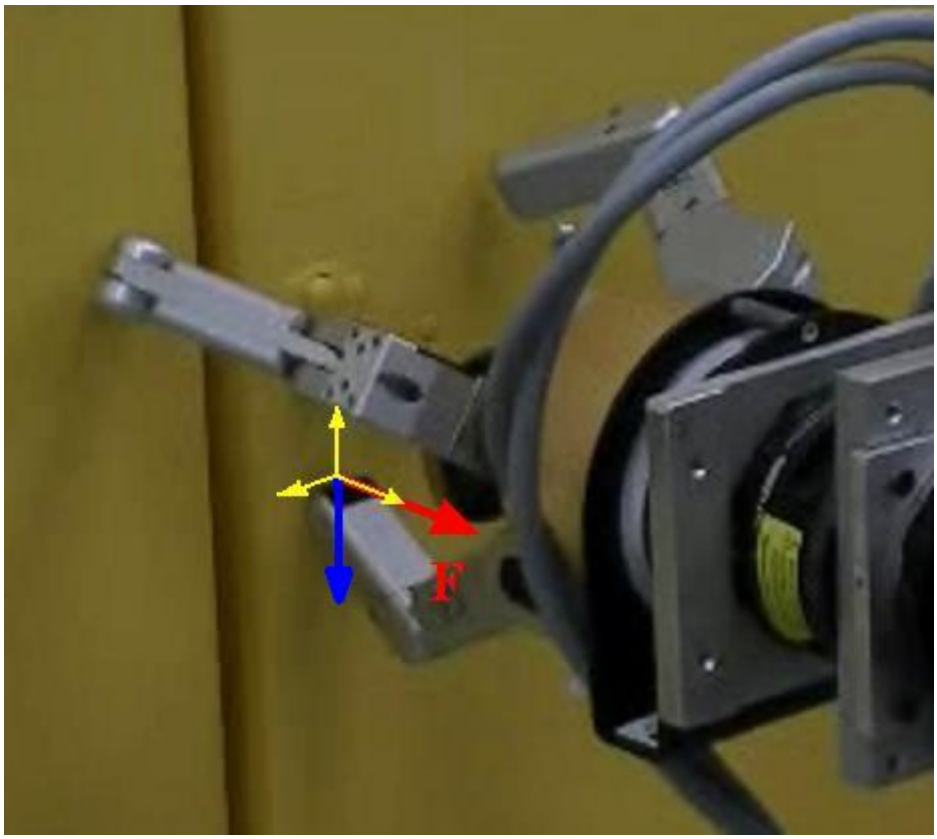


Figure 6-8: End effector in contact with the door, approaching the latch

Contact with the latch is identified when the vertical force (blue vector) exceeds a particular magnitude. After contact with the latch has been established, a force of 3lb, sufficient to lift the latch, is exerted and maintained. This force will be maintained into the next step, when the door is pulled ajar. The force required to actuate the latch (3lb) is determined by measuring the force while a human pushes on the fingertip with a force which is reasonable for lifting the latch.

6.2.2.5 Pull door ajar

In this step, the door is pulled open a few degrees. In addition to the obvious benefit to the task, opening the door, force data is used to identify the location of the door hinge for use in the next step. While pulling the door ajar, the following principles from Chapter 5 are demonstrated.

- 5.3.1 "Threshold monitoring" – forces are monitored and the algorithm reacts if a threshold is exceeded
- 5.4.1 "Force as input to position controller" – opening the door
- 5.5.3 "Identifying geometric features" – identifying the hinge location

During this step, the forces are monitored. If they exceed 4lb the manipulator stops and the operator is notified of the excessive force. It is then left to the operator to determine the best way to respond.

The door radius is unknown beforehand, so the manipulator pulls away from the door perpendicular to the face of the door. The latch is constrained to follow an arc centered about the hinge while the manipulator is pulling in a straight line. As the distance between the arc and the straight line increases, so does the lateral force on the end effector as shown in Figure 6-9.

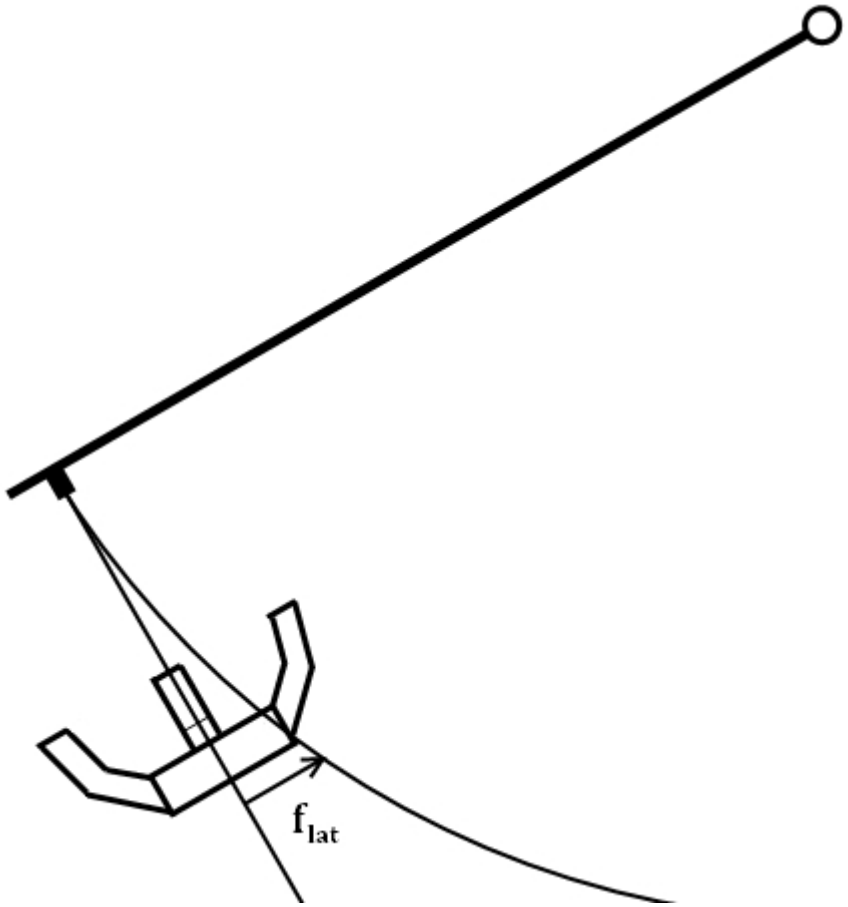


Figure 6-9: Lateral force while opening door

By minimizing this force, the error between the hand position and the latch position is minimized.

$$^{S}f_{y} = 0 \tag{6-13}$$

The hand approximately follows the arc of the latch. As the manipulator is pulling the door open, the Global *x* and *y* coordinates are recorded. At the end of this brief pulling step, a circle is fitted to these points, as shown in Figure 6-10. The center of the circle is the

location of the hinge in the Global Frame. By fitting a circle to this arc, the center of the circle (the hinge) and the arc radius can be determined and used in the last step.

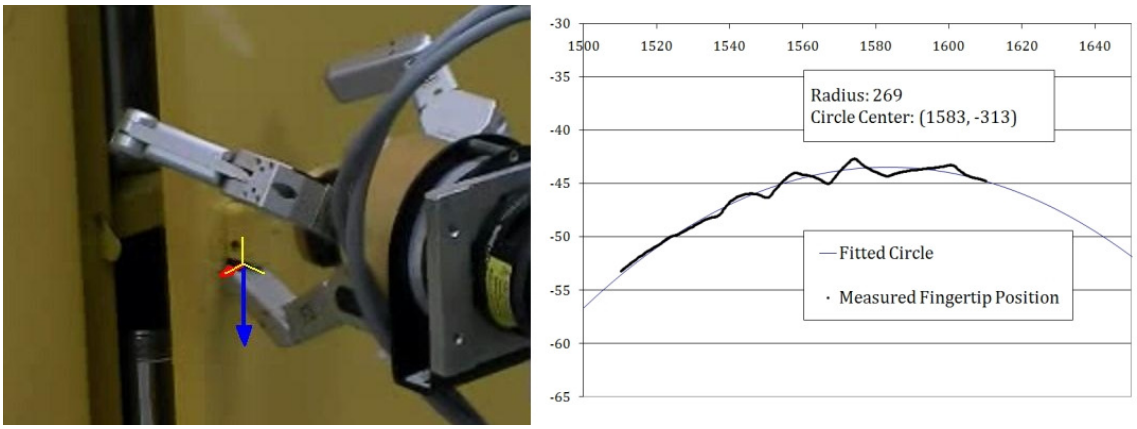


Figure 6-10: (Left) Pulling the door ajar; latch force as seen by sensor (blue vector) pointing down and lateral force (red vector) pointing left. (Right) Measured fingertip *x* and *y* position in Global coordinates with fitted circle

6.2.2.6 Push door open to 90 degrees

Having identified the location of the hinge, the door can be pushed open by reaching behind the door and following an arc centered at the hinge. In this step, only Section 5.3.1 "Threshold monitoring" is demonstrated.

Throughout this step, the forces are monitored. If they exceed the safety threshold of 4 pounds, the manipulator stops and the operator is notified. The operator is then allowed to take the proper course of action. The most likely source of excessive force in this step is an obstruction or geometric irregularity on the backside of the door.

The hand moves behind the slightly-opened door. The distance from the starting point behind the door to the hinge is the radius the hand will follow. The end effector

position is then commanded in Global coordinates based on this position. The manipulator stops pushing the door open when it has been opened 90 degrees.

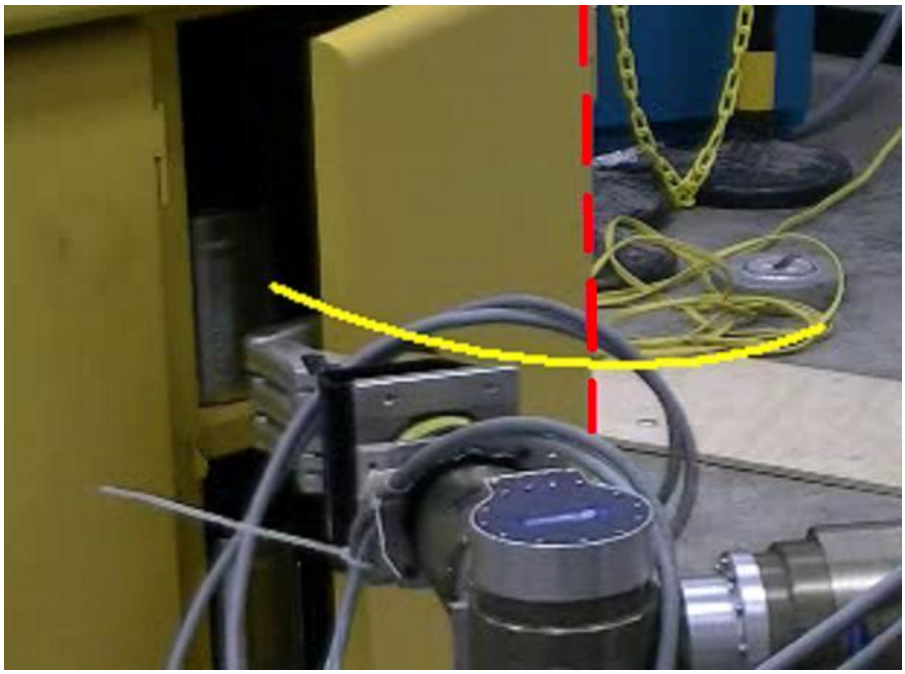


Figure 6-11: Pushing the door open following an arc about the estimated hinge location

The definition of the dot product is used to determine the angle which the door has been opened.

$$\theta = \cos^{-1}\left(\frac{A \bullet B}{|A||B|}\right) \tag{6-14}$$

Because the Workpiece Frame *x*-axis has been oriented to the face of the closed door, it is used as the baseline vector, A. The vector from the hinge to the current end effector location is roughly equal to the angle which the door is open (errors are due to the

thickness of the door, the estimated point on the hand making contact, and manipulator position tolerances). Substituting into equation 6-14:

$$\theta = \cos^{-1}\left(\frac{{}^{G}_{W}\mathbb{R}\hat{\mathrm{x}} \bullet \left[{}^{G}\rho_{EEF} - {}^{G}\rho_{hinge}\right]}{\left\|\left[{}^{G}\rho_{EEF} - {}^{G}\rho_{hinge}\right]\right\|}\right) \tag{6-15}$$

Because the end effector is following an arc centered at the hinge, the magnitude of the vector from the hinge to the end effector remains constant and is equal to the current radius.

$$\theta = \cos^{-1}\left(\frac{{}^{G}_{W}\mathbb{R}\hat{\mathrm{x}} \bullet \left[{}^{G}\rho_{EEF} - {}^{G}\rho_{hinge}\right]}{radius}\right) \tag{6-16}$$

By checking this condition, the algorithm can be stopped when the angle is greater than 90° (or any desired open angle).

6.2.3 Summary

The cabinet door is opened, demonstrating the three uses of force and torque data suggested in Chapter 5. The demonstration uses a six-axis force sensor to collect the force data. The sensor data is transformed to and from frames of interest like the points of contact and the End Effector Frame. In these frames, the forces have physical meaning, like the force being applied to the latch, or abstract meaning, like the direction to move the end effector while following the arc of the door latch.

More generally, this demonstration includes force data to improve the operational capability of kinematically controlled manipulator in the three objective areas which drastically reduced the burden on an operator performing the task using tele-operation. First, safety was improved by continually monitoring the contact forces. Second, the world model initially specified using vision was refined using force data to confirm and refine the location of the cabinet, latch and radius of the door. Finally, contact forces were managed to ensure the latch release mechanism was successfully released without generated excessive force on the gripper or environment.

6.3 RRC Collision Detection

6.3.1 System Setup

In this section a 7-DOF Robotics Research Corporation (RRC) K1207i robotic manipulator equipped with joint torque sensors will be used to demonstrate improved safety by a collision detection algorithm. The robot used is the same one presented in Section 1.3.3. The robot is placed inside a six-foot diameter sphere as shown in Figure 6-12.

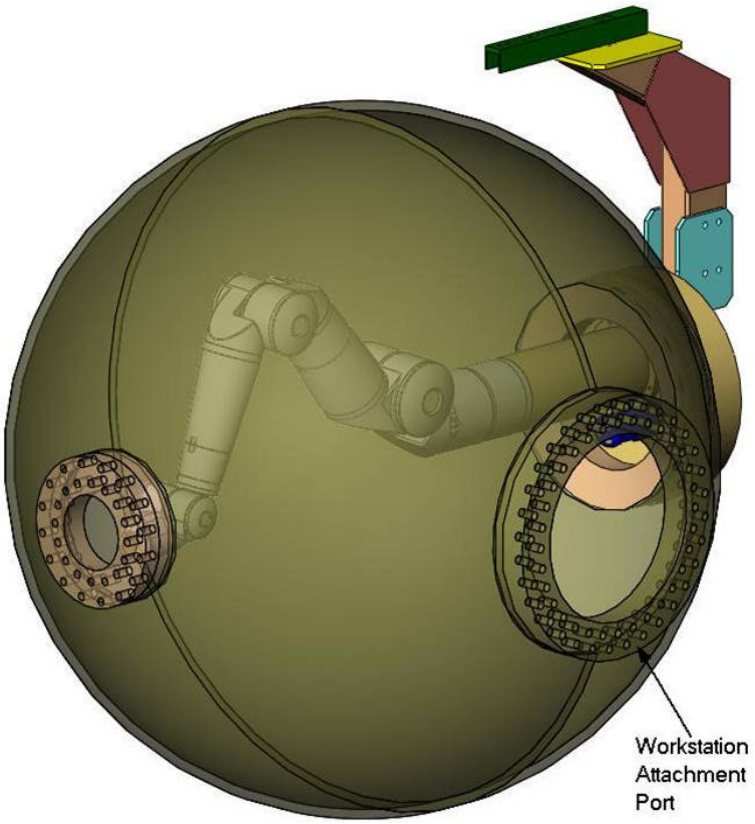


Figure 6-12: RRC K1207i in sphere to be cleaned

The methods discussed in Section 5.2.5.2 are used to isolate the forces due to contact from those due to internal sources. Next, the measured forces during free motion are compared to the computed values to validate the model and determine threshold values. Then sensor-based collision detection is demonstrated.

6.3.2 Sensor calibration and contact force isolation

The sensors were calibrated using the method described in Section 5.2.3.2. The sensor for the 7th joint was not calibrated. To calibrate that joint would require an external mass, such as a tool, to be attached after the last link. There are several end effector tools in testing and development stages. These were not included because the tools were not

available for this research. The lack of a tool has no unique import on the ability to demonstrate collision detection.

The gains for joints 1, 3, and 5 are all calculated with the manipulator at joint positions [-90, 90, 0, -90, 0, -90, 0]. The manipulator, as viewed from above, is diagramed below.

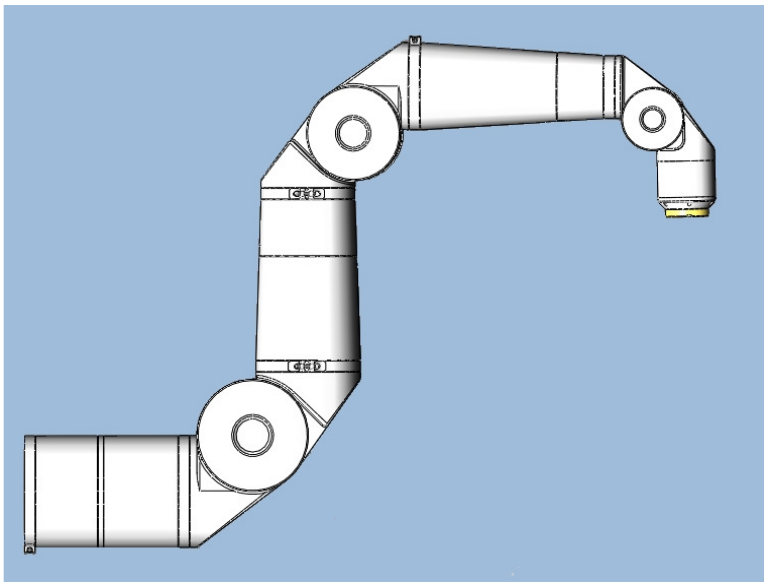


Figure 6-13: Top view of manipulator configuration during gain characterization for joints 1, 3, and 5

A mass[3] was hung from the end of the last link; a distance of 4.8 inches from joint 6. This is shown in the graphs for joints 1, 3, and 5 in Figure 6-14 from $0.5 \leq t < 0.75$ minutes. The empty bottle was placed back on the manipulator (Figure 6-14) after $t \approx 1.7$ minutes so the mass of the bottle could be subtracted from the mass of the bottle with water.

[3] Full 1-liter water bottle

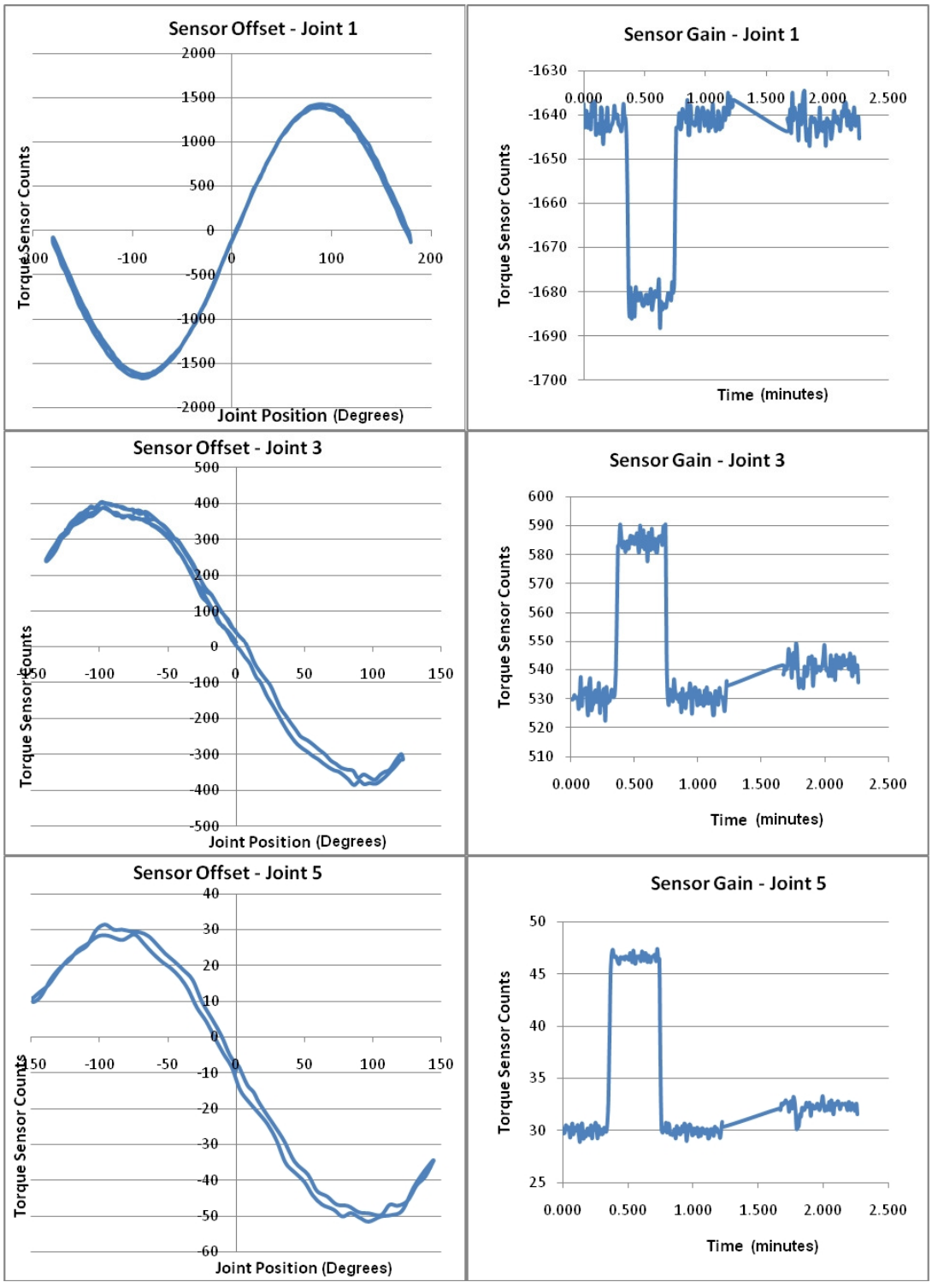


Figure 6-14: Sensor offset and gain calibration data for joints 1, 3, and 5

The gain for joints 2, 4, and 6 were calculated with the manipulator positions at [0, 90, 0, -90, 0, 0, 0]. Figure 6-15 shows the manipulator from the side during calculation of gain for joints 2, 4, and 6. For joints 2, 4, and 6, the joint limits prevented motion through a full 360°. In these cases the offset had to be estimated using a different technique. The period is known to be 1 cycle per 360°. This is physically evident and reflected in the graphs for those joints whose range is greater than 360°. So, the midpoint of the curve was determined by finding the minimum (or maximum) number of counts (derivative is zero) then moving 90° before or after that point. The slope of the torque was used as a check. Where the slope was at a max or min, there should also be the center of the sine wave (i.e. the measurement at that point should be zero). For determining the gain of joints 2, 4, and 6, the mass was attached approximately $1 < t < 2$ minutes. The empty bottle was hung back on the manipulator for $t > 4$ minutes.

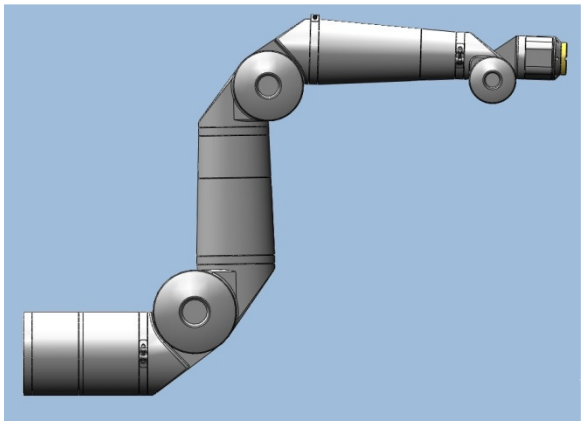


Figure 6-15: Side view of manipulator during gain calculation for joints 2, 4, and 6

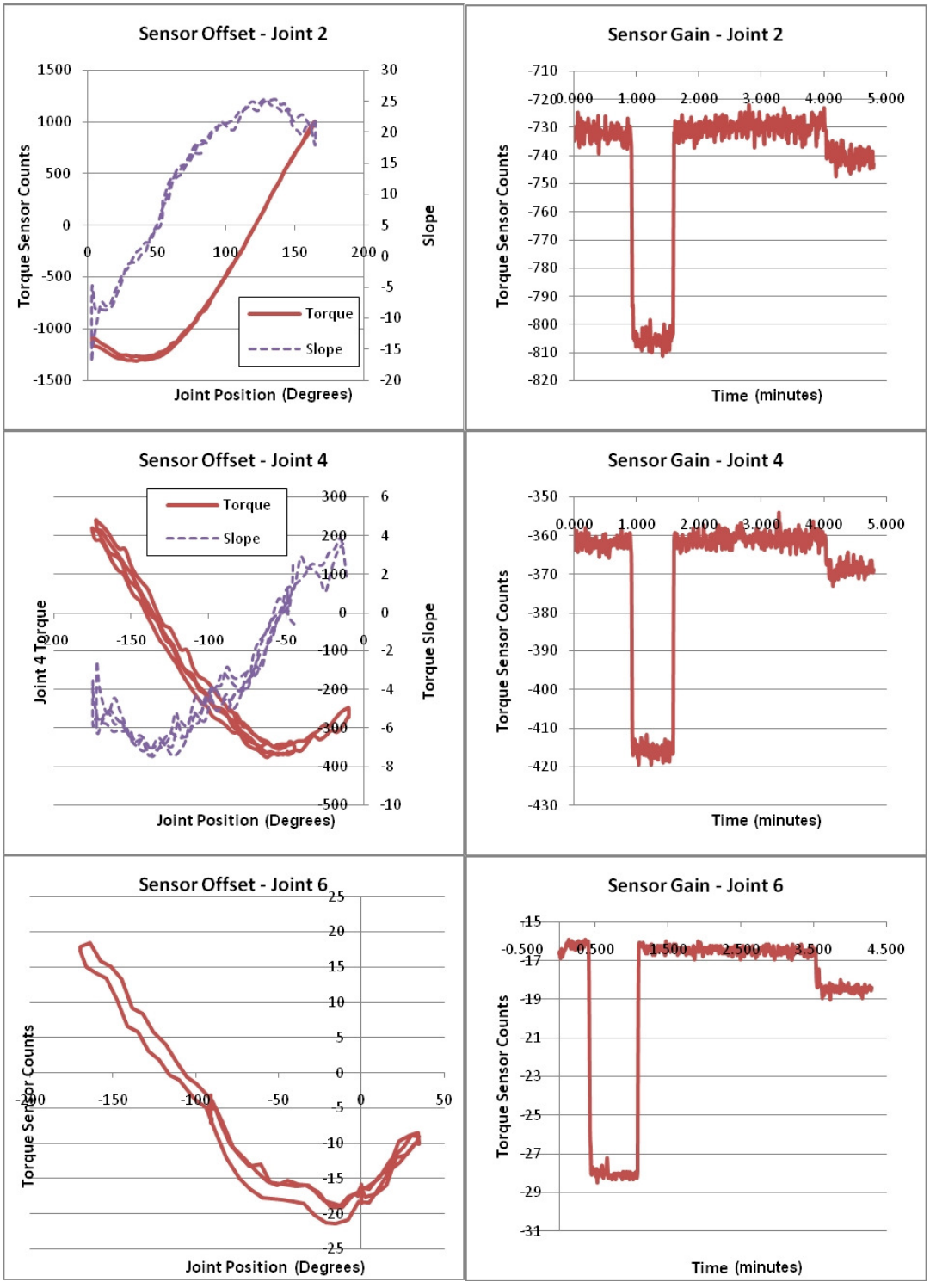


Figure 6-16: Sensor offset and gain calibration data for joints 2, 4, and 6

The final gains and offsets are shown in the table below.

Table 6-1: Torque sensor gains and offsets

	Gain [in-lb/count]	Offset [in-lb]
Joint 1	-1.3	-150
Joint 2	-1.23	120
Joint 3	-1.376	-13
Joint 4	-1.337	40
Joint 5	-1.55	-18
Joint 6	-1.55	2.5
Joint 7	-1	0

By applying the gains and offsets, per equation 5-1, the readings taken from the torque sensors are converted into meaningful torque measurements. These measurements are used in the next section, Section 6.3.3, to validate the model calculated in OSCAR and establish thresholds. In Section 6.3.4 the model and torque measurements are used to detect collisions.

6.3.3 Manipulator model

Collisions are detected by identifying torques from external sources, i.e. from contact with the environment, as shown in equation 4-28. The total torque is measured by the sensors and the torque due to internal sources is predicted by the manipulator model.

6.3.3.1 Model validation

The model must be validated before it can be used. A certain level of accuracy is required. With an inaccurate model collisions can be missed or falsely detected. The dominant internal source of torque in the RRC K1207i at its relatively slow operating speeds is due to its own weight. Given the slow operational speeds, the dynamic torques

due to acceleration, Coriolis, and centripetal effects could be neglects. This is generally known as the *quasi-static* assumption. It is validated by comparing the measured values to the predicted torques due to gravity alone.

Mass properties (mass and center of mass) provided by RRC are shown in Table 6-2. OSCAR is used with this robot for other tasks besides this experiment. Note, the mass of link 1 is obviously not zero but no actuator or sensor exists before that link. It is essentially connected to the ground. Its mass doesn't affect the joint torque model.

Table 6-2: Link mass and center of gravity data

Link	Distance to CG [mm]			Mass [kg]	Weight [lb]
	X	Y	Z		
1	0	0	0	0	0
2	0	0	0	19.051	41.9122
3	0	0	-222.25	9.299	20.4578
4	0	0	0	11.249	24.7478
5	0	0	-226.06	5.897	12.9734
6	0	0	0	4.536	9.9792
7	0	0	-55.8	2.403	10.5732

To determine the validity of the model, it is first compared to hand-calculations for different manipulator configurations relative to the expected mass properties. Next, the model is compared to the measured values in the absence of any external torques. Both stationary and dynamic measurements are taken to verify the model's accuracy without the Coriolis and inertial effects. The model is calculated using OSCAR's Dynamics classes.

6.3.3.1.1 OSCAR-Hand calculation validation

The OSCAR model is compared to hand calculations to ensure that the conversion from the units provided by OSCAR could be converted to known units. The calculations are done with a stationary manipulator. This accounts for the gravitational part of the model. The insignificance of the Coriolis and inertial torques is confirmed in the next section.

The expected torques are calculated for the same configurations used in Section 6.3.2. Each joint torque in each configuration is shown below.

Table 6-3: Comparison of OSCAR torques to hand-calculated expected torques

Joint	Configuration 1 (Figure 6-13)		Configuration 2 (Figure 6-15)	
	Expected Torque [in-lb]	OSCAR Torque [in-lb]	Expected Torque [in-lb]	OSCAR Torque [in-lb]
1	1899.9	1900.1	0	0
2	0	0	991.8	991.8
3	-684.3	-684.3	0	0
4	0	0	518.2	518.2
5	-61.0	-61.0	0	0
6	0	0	25.5	25.4
7	These torques are not calculated because the experiments are performed without an end effector tool			

6.3.3.1.2 OSCAR-Measurement validation

Next OSCAR's predicted values were compared to the measured values. Comparisons between the measured and computed gravity torques were performed while the manipulator was moving. Graphs of the comparisons are shown in Figure 6-17.

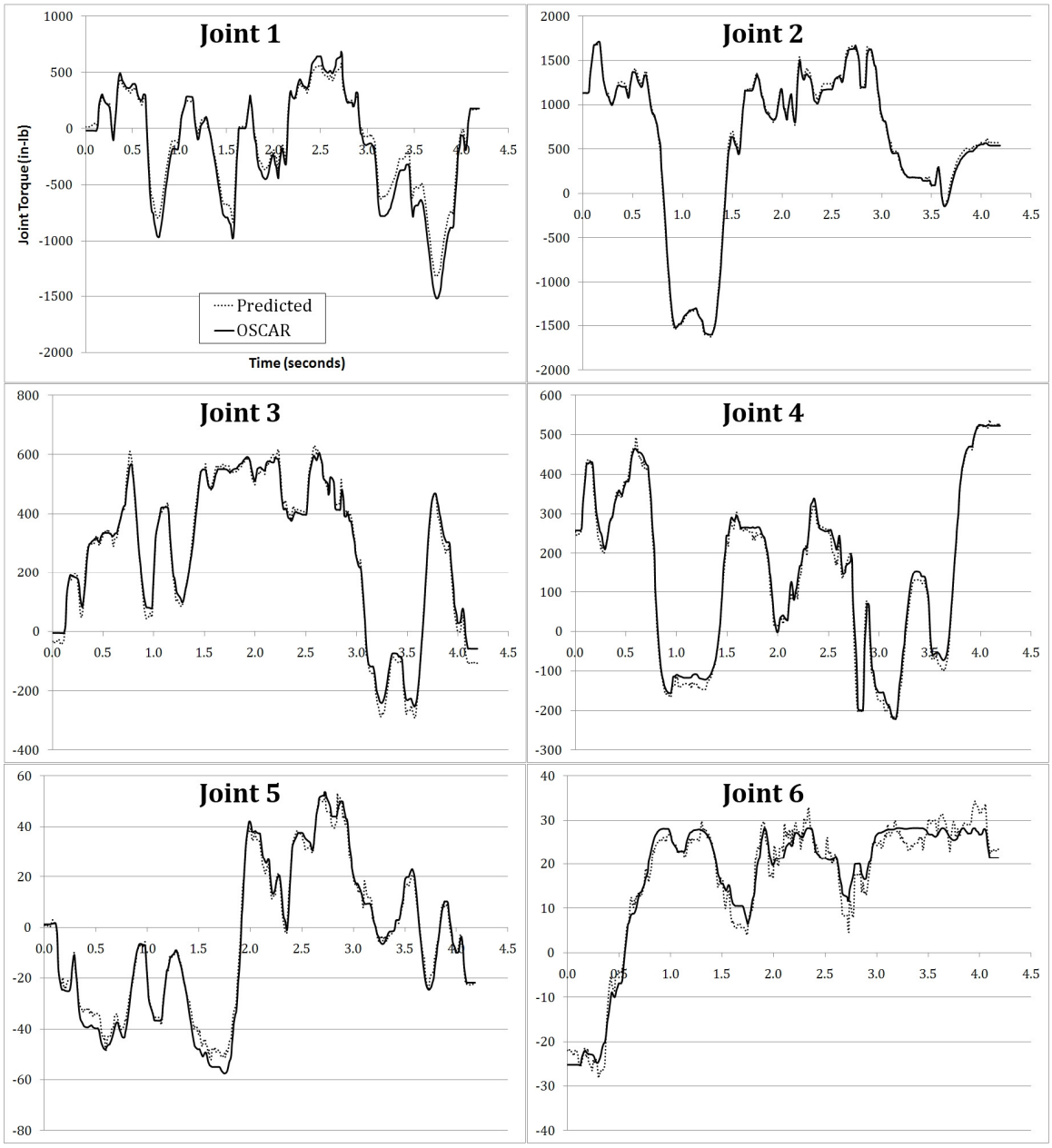


Figure 6-17: Comparison of OSCAR calculated torques to measured torques for various free-space motions in end effector space

6.3.3.2 Establishing thresholds

As discussed in Section 5.3.2.1, if the algorithm flagged a collision every time the measured torque differed from the predicted value, the controller would constantly flag collisions due to model deficiencies and signal noise.

$$\tau_{total} = \tau_{int} + \tau_{ext} + \tau_{uncertainty} \tag{6-17}$$

For the demonstration of collision detection, thresholds were chosen to be nearly the same as the maximum error experienced during the experimental end effector motions. The maximum errors and chosen thresholds are recorded in Table 6-4.

Table 6-4: Maximum error and selected error threshold for joints 1 through 6

	Max Error	Thresholds
Joint 1	213.20	200
Joint 2	119.58	100
Joint 3	60.90	60
Joint 4	94.66	90
Joint 5	8.64	10
Joint 6	9.40	10

The thresholds can be adjusted depending on the desired sensitivity of the collision detection scheme. If a highly-sensitive system is needed; one in which all of the collisions are detected, the thresholds can be lowered. Conversely, if the system should be more insensitive to collisions, the thresholds may be increased. The threshold should be chosen to protect the robot and ensure the task can be completed.

6.3.4 Collision detection

The collision detection algorithm presented in Section 5.3.2 is used on the RRC arm to detect collisions. The thresholds determined in Section 6.3.3.2 are used to detect the collisions. To test the algorithm, the joint positions, OSCAR-predicted torques, and measured torques are recorded. The position and torque sensors provide measurements at roughly 50Hz. Every ten measurements are averaged to create one data point so that a data point is generated roughly every 0.5 seconds. A collision is recorded if the magnitude of the difference between modeled and measured torques exceeded the threshold value for any joint.

```
If(abs(torque_measured-torque_OSCAR)>torque_threshold)
{
    Collision = true;
}
```

The measured and predicted torques are plotted. The threshold is subtracted and added to the modeled value to indicate the range within which the measured torque must stay in order not to trigger a collision. Collisions were induced by applying a force of unknown magnitude and direction for an unknown duration at an unknown location on the manipulator. Most collisions occurred after the third joint. Because this is a testing stage, collisions are recorded manually by clicking a button during the recording. Collisions are shown on all 6 graphs as a red "X". Because it is impossible to manually jog the robot and indicate a collision simultaneously, collisions induced during motion are manually marked briefly after the collision. Each graph also indicates if the difference between measured and modeled torques exceeds the threshold for ANY of the joints and if it exceeds the threshold for that particular joint. A general collision detection is indicated by an "O". A collision

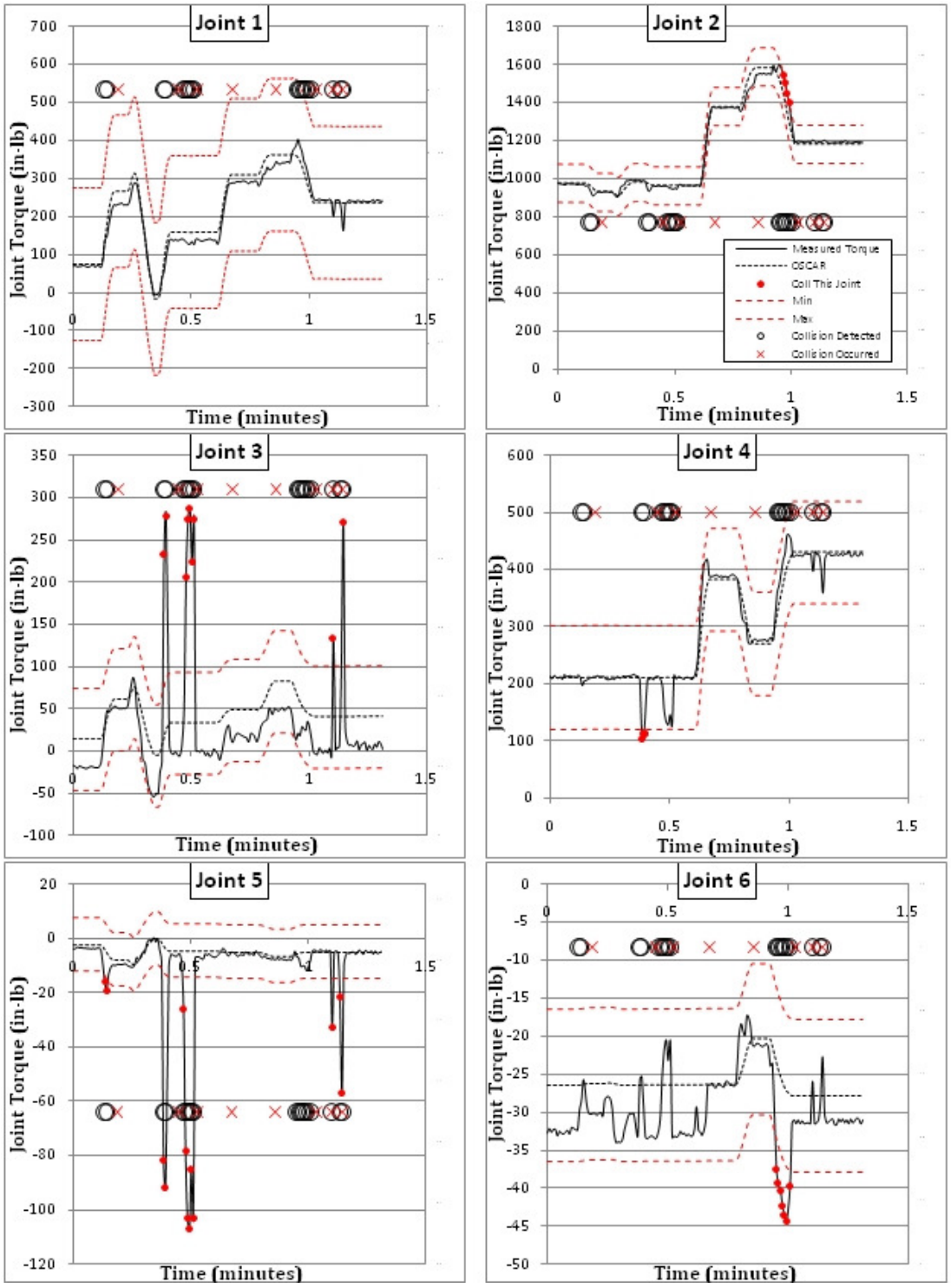


Figure 6-18: Collision detection graphs

which is detected in that particular joint is indicated by a red dot on the measured joint torque line. Sample graphs for all of the joints are shown in Figure 6-18.

Noting the changing predicted torque in at least one joint at the time of collision, it can be seen that before $t \approx 1.0$ minute, the manipulator was moving during each collision. During the last two collisions, the manipulator was stationary. The collision detection was successful 6 of 8 times, missing the collisions at $t \approx 0.7$ and $t \approx 0.8$ minutes.

The first of these missed collisions is difficult to see in any of the graphs. The only two joints where the collision is visible are joints 3 and 4. It is likely that this force was too small to affect much change in joint torque and/or was in a direction which did not induce much torque in the joints. One of the spikes in joint 6 may be attributed to this collision as well, but the torque in joint 6 changes erratically throughout the experiment.

Evaluated absolutely, the threshold for joint 6 (± 10 in-lb) is small. Given the range of predicted torques for joint 6 ($-30 < \tau_6 \leq -20$ in-lb), this range is quite large. The tuning for joint 6 doesn't appear as good as the other joints (Figure 6-17). It is difficult to identify the importance of this threshold. Collisions are still detected in this joint (around $t = 1$ minute).

At least one collision was detected by each joint except joint 1. It cannot be determined from the data if a different manipulator configuration or point of contact or direction or magnitude of force may have triggered a collision in joint 1. It is probable, however, because the last two collisions can be seen as small, brief deviations of the

measured torque from the predicted for joint 1. With a large enough collision force, the spikes in torque would likely exceed the threshold.

To provide some insight into the collision detector performance, the maximum force which might go undetected in each principle direction (*x*, *y*, and *z*) was calculated. These forces were generated by calculating the torque in each joint given a 1lb force in the given direction at the end effector.

$$\tau_{ext}^{x,1} = \left[G_{\phi}^{e} \right]^{T} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{6-18}$$

Now, each side of the equation is scaled by a constant until the torque at each joint, i, is equal to twice the threshold for that joint.

$${}^{i}K\,{}^{i}\tau_{ext}^{x,1} = 2 * {}^{i}\tau_{threshold} = {}^{i}K \left[G_{\phi}^{e} \right]^{T} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{6-19}$$

The constant for each joint represents the maximum force which might go undetected by that joint. So the maximum force for the entire manipulator is the minimum of these[4]. The maximum collision force which might not be detected for each of the principle directions for a collision at the end effector are shown in Figure 6-19.

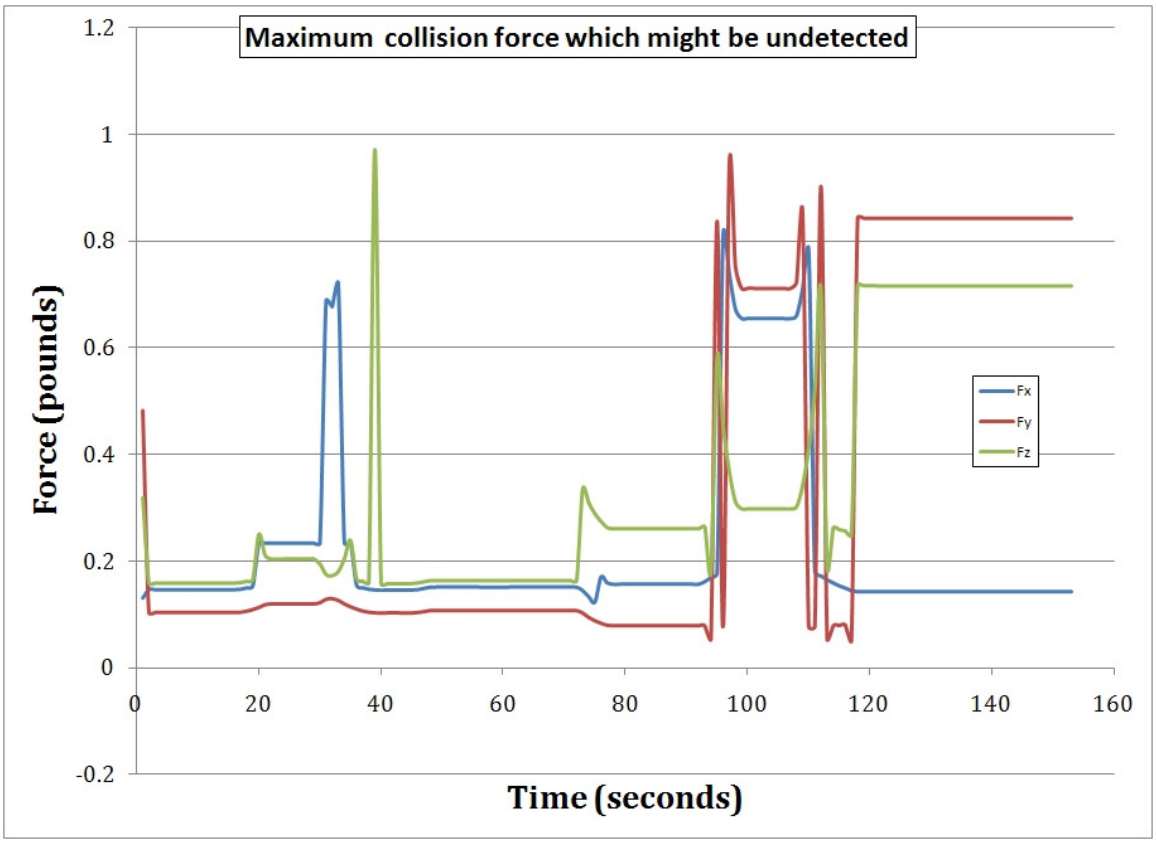


Figure 6-19: Maximum forces at end effector which might go undetected

It should be noted again that the calculations were for collisions at the end effector only and only in the principle directions. Changing the direction of force may result in

[4] If the force is in one of the principle directions and exceeds more than one of the constants, ${}^{i}K$, then the force will be detected by all those joints for which ${}^{i}K$ is less than the force.

different forces. Also, the sample calculation does not include collisions at other locations on the manipulator. As the distance from the joint to the collision changes, so will the force.

For the moves executed in this test, the maximum collision force which might not be detected doesn't exceed 1 lb at the end effector. The RRC K1207i has a maximum continuous-duty payload of 35 lb so for collisions at the end effector, the algorithm with the presented assumptions and tuning performs quite well.

6.4 Chapter summary

In this chapter, all of the proposed uses of force and torque data from Chapter 5 were demonstrated. All three, 5.3 "Safety", 5.4 "Managing contact forces using positional control", and 5.5 "World Model Augmentation" were demonstrated with the six-axis sensor while opening the cabinet. Improved safety, in the form of collision detection (section 5.3.2), was demonstrated using joint torque sensors on the RRC arm. The maximum force which might not be detected at the end effector does not exceed 1 lb so the collision detection algorithm performs quite well.

7 Conclusions and future work

7.1 Conclusions

In the introduction, several manipulators were presented which are typical of the robotics applications currently being used in nuclear and radioactive environments. A few potential benefits of force and torque data were outlined there.

- In the ARIES system, force and torque data could be used to identify collisions and assist in manipulation.
- A six-axis force sensor mounted at the wrist of the Mighty Mouse robot could have verified contact with the environment, guided tool usage, and prevented excessive forces.
- Joint torque sensors in the RRC arm can be used to detect collisions, as demonstrated in Chapter 6, and may even be used to determine and control contact force during manipulation.

In Chapter 5, the technology required for the above benefits were discussed in a generic nature. The principles presented in Chapter 5 can be quickly adapted to other manipulators, sensors, and applications. This is demonstrated in Chapter 6 by applying the algorithms to two different manipulators using two different force sensor types.

When opening the cabinet door using the LWA3 and the ATI six-axis force sensor, all the benefits proposed in Chapter 5 were adapted to the specific task. The RRC arm and the joint torque sensors were used to detect anomalous forces for collision detection. The algorithms can be applied to other tasks or they can be used to provide feedback to the controller or the operator. Force limits were demonstrated during an automated procedure

but they can be adapted to run while in a teleoperation mode. Similarly, collision detection, demonstrated with the RRC arm, can be performed during an automated procedure.

7.2 Future work

Future work can be divided into two areas: improving the performance of the demonstrated algorithms and extending the operational capabilities of these algorithms.

7.2.1 Demonstration improvement

The following suggestions could improve the performance of the algorithms without requiring new research. They were left out of this work to demonstrate the effectiveness of the algorithms given a very reasonable investment of time and resources.

7.2.1.1 Cabinet demonstration

The following improvements are focused on the performance of the cabinet-opening demonstration.

7.2.1.1.1 Generalization

Some of the details of the door opening demonstration are specific to the type of door and type of latch. The current algorithm only opens latches which are operated by lifting. However, similar algorithms could be implemented to open a latch which must be turned (e.g. a doorknob). The first implementation would be to allow the operator to choose the correct latch type from a list and have the robot open the latch. Next, the robot may try different latch opening schemes and determine the correct type online using force and position feedback for failure criteria (e.g. attempting to turn a locked doorknob – by

comparing the change in position to the change in force, a human recognizes that the door is locked).

The door in this demonstration is opened by pulling. It swings to the robot's right. The current algorithm assumes the hinge is to the robot's right of the latch. This assumption can be relaxed by a simple logical check after the process of pulling the door ajar and determining the location of the hinge. If the hinge is located to the left of the latch position, the door opens left. If the hinge is to the right, it opens right.

Additionally, the type of door could also be chosen by the operator from a list of potential doors. A different algorithm would be executed if the operator identifies the door as a slide or push type door. As with the latch, the next step would be for the manipulator to attempt different methods until the correct type is determined. Similar to a human who approaches a door and pushes when it should be pulled (except the robot won't get embarrassed).

The generalized door opening method (complete with latch actuation) could be put into a robotics library like OSCAR. One possible design might be a Door Opening class. A template would exist for Door Types and Latch Types. The Door Opening class might contain a pointer to Door Types. This pointer could point to a vector populated with instances of the Door Type template corresponding to different types of doors. Another pointer would point to a similar set of Latch Types. A new door or latch type could be added by creating a new class from the corresponding template.

7.2.1.1.2 Data filtering

Using the same filter design, the number of elements in the filter may be increased or decreased to improve the performance of the filter. Other filtering designs exist and may better reduce the noise in the measurement.

7.2.1.1.3 Controller design

The parameters of the controller were chosen by trial and error. Analysis and design of the controller may yield better performance, especially in the second step when the hand rotates to make contact with the face of the cabinet. The simple contact model used in the thesis could be used, but the modeling techniques mentioned later in Section 7.2.2.2.1 may also be helpful. The better the model is, the better the potential for control.

7.2.1.2 RRC collision detection

For the RRC collision detection demonstration, the major areas of improvement are the sensor calibration and the manipulator model. More room for improvement probably exists with the manipulator model because the largest uncertainties in the torque occurred when the manipulator was moving quickly or accelerating (some overshoot can be seen when the manipulator stops quickly).

7.2.1.2.1 Sensor calibration

The sensor calibration in Section 6.3.2 was performed offline for simplicity. The method presented by Ma [1994] allows online calibration. These algorithms could be adapted to OSCAR for future calibration on other manipulators. This would expedite this necessary and rather frustrating portion of the setup for future applications.

The calibration for the joint torque sensors was sufficient to provide adequate collision detection but the accuracy could be improved. The torque applied to the joints during gain calibration was potentially inaccurate. It was estimated from the mass of one liter of water. The volume of water was measured by filling a commercially-available hiking water bottle. Based on the marking on the bottle, it was assumed 1 liter of water was present. This was assumed to be equal to 1 kilogram. Using a calibrated mass and a scale to apply the torque would yield more accurate calibration.

7.2.1.2.2 Manipulator model

As mentioned before, joint torques due to Coriolis and inertia were omitted in this work to demonstrate the possible performance with moderate investment of setup time. Adding these parts to the torque model would require determination of the inertial parameters of each of the links. Attempts were made, both by estimating the links as uniform density solid bodies and by published methods. Roughly 3-4 days were committed to acquiring these parameters but resulted in absurd calculations of joint torque.

7.2.2 Algorithm development: future work

Whereas the previous recommendations were focused on improvement of the demonstrated algorithms in these particular applications, the following recommendations are focused on current limitations of research in the area.

7.2.2.1 Safety

7.2.2.1.1 Collision detection performance

As discussed in Section 5.3.2.2.2, the performance of a joint torque sensor collision detection algorithm is not obvious. Current performance analysis is demonstrative; detecting collisions before damaging delicate objects such as balloons [De Luca, Albu-Schaeffer, et al., 2006] or even humans [Haddadin, et al. 2008].

A quantitative method for comparing the performance of two methods or determining if an algorithm meets minimum safety criteria doesn't exist. Such a rating may determine the maximum collision force which may go undetected at critical locations on the manipulator or in the workspace. This would be difficult because any point on the manipulator may experience the collision and in certain parts of the workspace, any number of links may pass through.

7.2.2.2 Force as input to position control

7.2.2.2.1 Contact modeling

There has been a significant effort in contact modeling external to our group. In the demonstrations above, world-model augmentation was application specific and the integration or more generalized approaches would likely require the efforts of a researcher dedicated to the task A brief reveals several promising approaches. [Baraff, 1989], [Baraff, 1990], etc. From these articles, a better model of the relationship between the contact force and the displacement could be developed. This improved model could also aid in designing a faster controller (Section 7.2.2.2.2).

7.2.2.2.2 Controller analysis and design

The contact-level controller used in the cabinet demonstration doesn't need to push the current boundaries of research, but it could be improved. The current controller is proportional only. A better model of the contact (Section 7.2.2.2.1) may show that a proportional controller is not the best choice for stability and steady state error. Additionally, the control parameters could be designed for the best response time without overshoot.

7.2.2.2.3 Calculation of end effector force using joint torque sensors

It is theoretically possible to calculate the end effector force from the joint torques. This measurement will be subject to the same errors associated with collision detection. The calculated force at a given point could only be determined within a particular range of forces. It would be interesting to determine how effectively the calculated force can be determined for use in future algorithms. If it can be accurately calculated, the calculated force could be used in the same way that the sensed force of a six-axis force sensor is used. The measured force could be used to determine a new end effector position. Because the joint positions are set, there should be no issue calculating the end effector force for a given vector of joint torques.

7.2.2.3 World model augmentation

7.2.2.3.1 Feature identification

In this thesis, the cabinet door was assumed to be flat. The feature identification performed using vision and force data, could be generalized. An acceptable threshold for

the R^2 value of the line fit to the data could be established as a threshold. If the R^2 value for a particular fit is less than the threshold, the next order curve is test-fit to the data. This could be repeated, raising the order of the fit curve, until an acceptable R^2 value is found.

Research has been done in this area for identification based on data collected by visual sensors. It would be interesting to apply that research to tactile identification. That research may also help with distinguishing one object from another in a point cloud, be it generated by vision or touch.

7.2.2.3.2 Model integration

Research is being done in the Nuclear Robotics Group on world modeling. The world model information gathered from the algorithms in this thesis should be added to the world model presented in the thesis of Brian O'Neil.

7.3 Chapter summary

Several ideas for future work are presented in this chapter. The first suggestions can improve the performance of the demonstrations. These are mostly focused on continuing work which is inside the boundaries of current research. Future research work is also proposed. These suggestions would expand research boundaries to increase understanding in areas surrounding the work presented in this thesis.

References

An, Chae H, Christopher G Atkeson, and John M Hollerbach. "Estimation of Inertial Parameters of Rigid Body Links of Manipulators." *IEEE International Conference on Decision and Control.* Ft. Lauderdale, FL: IEEE, 1985. 990-995.

Baraff, David. "Analytical Methods for Dynamic Simulation of Non-penetrating Rigid Bodies." *Computer Graphics*, 1989: 223-232.

Baraff, David. "Curved Surfaces and Coherence for Non-penetrating Rigid Body Simulation." *Computer Graphics*, 1990: 19-28.

Borangiu, Theodor, Mitica Manu, Florin D Anton, Silvia Tunaru, and Anamaria Dogar. "High-Speed Robot Motion Control under Visual Guidance." *IEEE EPE-PEMC.* Portoroz, Slovenia, 2006. 519-523.

Bronisz, Lawrence E, and Pete C Pittman. *Virtual Reality and Telepresence Control of Robots Used in Hazardous Environments.* Report, Los Alamos, NM: DOE Office of Scientific and Technical Information, 1996.

Brumson, Bennett. *Robo Removal: Parts Finishing is Big Business for Robots: Robotic Industries Association.* August 17, 2007. http://www.robotics.org/content-detail.cfm/Industrial-Robotics-News/Robo-Removal:-Parts-Finishing-is-Big-Business-for-Robots/content_id/635 (accessed May 2010).

Brumson, Bennett. *What Is New with Robots in Press/Machine Tending and Material Removal: Robotic Industries Association.* December 7, 2009. (accessed May 2010).

Cox, Daniel J, Jacy M Legault, Cameron J Turner, and Delbert Tesar. "Automated Plutonium Processing Workcell Technology." *American Nuclear Society Proceedings of the 8th Topical Meeting on Robotics and Remote Systems.* Pittsburgh, PA: American Nuclear Society, 1999. 1-17.

Craig, John J. *Introduction to Robotics: Mechanics and Control.* 3rd Edition. Upper Saddle River, NJ: Pearson Prentice Hall, 2005.

Craver, W M, and Del Tesar. "Force and deflection determination for a parallel 3-DOF robotic shoulder module." *ASME.* 1990.

De Luca, Alessandro, Alin Albu-Schaeffer, Sami Haddadin, and Gerd Hirzinger. "Collision Detection and Safe Reaction with the DLR-III Lightweight Manipulator Arm." *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2006: 1623-1630.

De Luca, Alessandro, and Lorenzo Ferrajoli. "Exploiting Robot Redundancy in Collision Detection and Reaction." *IEEE/RSJ International Conference on Intelligent Robots and Systems.* Nice, France: IEEE, 2008. 3299-3305.

De Luca, Alessandro, and Raffaella Mattone. "Actuator Failure Detection and Isolation Using Generalized Momenta." *IEEE International Conference on Robotics and Automation*, 2003: 634-639.

De Luca, Alessandro, and Raffaella Mattone. "Sensorless Robot Collision Detection and Hybrid Force/Motion Control." *IEEE International Conference on Robotics and Automation.* Barcelona, Spain, 2005. 999-1004.

Denavit, J, and R S Hartenberg. "A Kinematic Notation for Lower-pair Mechanisms Based on Matrices." *Transactions of the ASME Journal of Applied Mechanics*, 1955: 215-221.

Duchaine, Vincent, Nicolas Lauzier, Mathieu Baril, Marc-Antoine Lacasse, and Clément Gosselin. "A Flexible Robot Skin for Safe Physical Human Robot Interaction." *IEEE International Conference on Robotics and Automation*, 2009: 3676-3681.

Ebert, Dirk M, and Dominik D Henrich. "Safe Human-Robot-Cooperation: Image-based Collision Detection for Industrial Robots." *IEEE International Conference on Intelligent Robots and Systems.* Lausanne, Switzerland, 2002. 1826-1831.

Eitner, Christian, Yuto Mori, Kei Okada, and Masayuki Inaba. "Task and Vision Based Online Manipulator Trajectory Generation for a Humanoid Robot." *IEEE-RAS International Conference on Humanoid Robots.* Daejeon, Korea, 2008. 293-298.

Engineering Talk. *Improved robotic force platform and controls: Engineering Talk.* April 27, 2000. http://www.engineeringtalk.com/news/tat/tat102.html (accessed May 2010).

Eom, K S, I H Suh, and S-R Oh. "Disturbance Observer Based Force Control of Robot Manipulator without Force Sensor." *IEEE International Conference on Robotics and Automation.* Leuven, Belgium: IEEE, 1998. 3012-3017.

Epson. *RC620+ Open Architecture PC Based Controller: PC Based Controls.* 2010. http://www.robots.epson.com/products/pccontrols.htm#RC520Controller (accessed May 2010).

Fanuc Robotics. "Fanuc Software Options." *Logismarket.* Fanuc Robotics. http://www.logismarket.co.uk/ip/fanuc-robotics-uk-ltd-software-options-295948.pdf (accessed 2010).

Freeman, R A, and Del Tesar. "Dynamic Modeling of Serial and Parallel Mechanisms/Robotic Systems: Part I -- Methodology, Part II -- Applications." *Trends and Developments in Mechanisms, Machines and Robotics, Proceedings 20th Biennial ASME Mechanisms Conference.* 1988. 7-27.

García, J Gámez, A Robertsson, J Gómez Ortega, and R Johansson. "Force and Acceleration Sensor Fusion for Compliant Robot Motion Control." *IEEE International Conference on Robotics and Automation.* Barcelona, Spain: IEEE, 2005. 2709-2714.

Haddadin, Sami, Alin Albu-Schaeffer, Alessandro De Luca, and Gerd Hirzinger. "Collision Detection and Reaction: A Contribution to Safe Physical Human-Robot Interaction." *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2008: 3356-3363.

Harden, Troy A, and Pete Pittman. "Development of a Robotic System to Clean Out Spherical Dynamic Experiment Containment Vessels." *American Nuclear Society EP&R and R&RS Topical Meeting.* Albuquerque, NM: American Nuclear Society, 2008. 358-364.

Harden, Troy Anthony. *Minimum Distance Influence Coefficients for Obstacle Avoidance in Manipulator Motion Planning.* PhD Dissertation, Austin: University of Texas at Austin, 2002.

Hashtrudi-Zaad, Keyvan, and Septimiu E Salcudean. "Bilateral Parallel Force/Position Teleoperation Control ." *Journal on Robotic Systems*, 2002: 155-167.

Hernandez, E. *Compliance Modeling for General Manipulator Structures.* PhD Dissertation, Austin: University of Texas at Austin, 1996.

Hooper, Richard. *Multicriteria Inverse Kinematics for General Serial Robots.* PhD Dissertation, Austin: University of Texas at Austin, 1994.

Hoshi, T, and H Shinoda. "Robot Skin Based on Touch-Area-Sensitive Tactile Element." *IEEE International Conference on Robotics and Automation.* Orlando, FL, 2006. 3463-3468.

Hudgens, J, and Del Tesar. "A fully-parallel six degree-of-freedom micromanipulator: kinematic analysis and dynamic model." *Proc. 20th Biennial ASME Mechanisms Conf.* 1988.

Aaron Hulse, Amit Kulkarni, Brian O'Neil, Mitch Pryor, and Chetan Kapoor, "Improved grasping strategies for flexible manufacturing and mobile manipulation," *Proceeding of the 2009 ANS Winter Meeting*, Washington DC, November 2009.

Intelligent Peripherals for Robots. *Compliance Devices: IPR Automation.* http://www.iprautomation.com/iso.htm#compliance (accessed May 2010).

Jain, Advait, and Charles C Kemp. "Behaviors for Robust Door Opening and Doorway Traversal with a Force-Sensing Mobile Manipulator." *RSS Manipulation Workshop: Intelligence in Human Environments.* 2008.

Je, Hwan-Jook, Jun-Young Baek, and Min-Cheol Lee. "A Study of the Collision Detection of Robot Manipulator without Torque Sensor." *ICROS-SICE International Joint Conference 2009.* Fukuoka, Japan, 2009. 4468-4471.

Jung, Seul, T C Hsia, and Robert G Bonitz. "Force Tracking Impedance Control of Robot Manipulators Under Unknown Environment." *IEEE Transactions on Control Systems Technology*, 2004: 474-483.

Kapoor, Chetan. *A Reusable Operational Software Architecture for Advanced Robotics.* PhD Dissertation, Austin: University of Texas at Austin, 1996.

Lee, Jay. "Apply Force/Torque Sensors to Robotic Applications." *Robotics*, 1987: 189-194.

Lee, M H, and H R Nicholls. "Tactile Sensing for Mechatronics -- A State of the Art Survey." *Mechatronics*, 1999: 1-31.

Lee, Sang-Hyuk, Young-Loul Kim, and Jae-Bok Song. "Torque Sensor Calibration Using Virtual Load for a Manipulator." *International Journal of Precision Engineering and Manufacturing*, 2010: 219-225.

Liu, Guangjun, Karl Iagnemma, Steven Dubowsky, and Guillaume Morel. "A Base Force/Torque Sensor Approach to Robot Manipulator Inertial Parameters Estimation." *IEEE International Conference on Robotics and Automation*, 1998: 3316-3321.

Liu, Zhengshi, Yong Wang, Enwei Chen, Li Wen, and Yunjian Ge. "A Method for Measuring Dynamic Performance Index of Robot's Multi-axis Wrist Force Sensor." *IEEE International Conference on Information Acquisition*, 2005: 170-174.

Lu, Shujun, and Jae H Chung. "Collision Detection Enabled Weighted Path Planning: A Wrist and Base Force/Torque Sensor Approach." *IEEE International Conference on Advanced Robotics.* Seattle, WA: IEEE, 2005. 165-170.

Lu, Shujun, Jae H Chung, and Steven A Valinsky. "Human-Robot Collision Detection and Identification Based on Wrist and Base Force/Torque Sensors." *IEEE International Conference on Robotics and Automation.* Barcelona, Spain, 2005. 3796-3801.

Ma, Donghai. *Autonomous Torque Sensor Calibration and Gravity Compensation for Robot Manipulators.* Master's Thesis, Montreal, Canada: McGill University, 1995.

Ma, Donghai, John M Hollerbach, and Yangming Xu. "Gravity Based Autonomous Calibration for Robot Manipulators." *IEEE International Conference on Robotics and Automation (ICRA)*, 1994: 2763-2768.

Maples, James A, and Joseph J Becker. "Experiments in Force Control of Robotic Manipulators." *IEEE International Conference on Robotics and Automation.* IEEE, 1986. 695-702.

Marks, Paul. *Robots with skin enter our touchy-feely world: New Scientist.* April 19, 2010. http://www.newscientist.com/article/mg20627566.800-robots-with-skin-enter-our-touchyfeely-world.html (accessed May 2010).

Matsumoto, Taishi, and Kazuhiro Kosuge. "Collision Detection of Manipulator Based on Adaptive Control Law." *IEEE/ASME International Conference on Advanced Intelligent Mechatronics.* Como, Italy, 2001. 177-182.

McKee, Steven. *Actinide Research Quarterly--ARIES at 10.* July 2008. http://arq.lanl.gov/source/orgs/nmt/nmtdo/AQarchive/1st_2ndQuarter08/page1.shtml (accessed May 2010).

Merrick. "Beryllium Oxide Glovebox." *Merrick Project Sheets.* October 11, 2010. http://www.merrick.com/images/uploads/project_sheets/2670.pdf (accessed 2010).

Middleton, R H, and G C Goodwin. "Adaptive Computed Torque Control for Rigid Link Manipulators." *IEEE International Conference on Decision and Control*, 1986: 68-73.

Morikawa, Sho, Taku Senoo, Akio Namiki, and Masatoshi Ishikawa. "Realtime Collision Avoidance Using a Robot Manipulator with Light-Weight Small High-Speed Vision Systems." *IEEE International Conference on Robotics and Automation.* Rome, Italy, 2007. 794-799.

Morinaga, Shinya, and Kazuhiro Kosuge. "Collision Detection System for Manipulator Based on Adaptive Impedance Control Law." *IEEE International Conference on Robotics and Automation.* Taipei, Taiwan, 2003. 1080-1085.

Ogando, Joseph. *Force Control and Machine Vision Guide Robots: Design News.* June 24, 2007. http://www.designnews.com/article/1147-Force_Control_and_Machine_Vision_Guide_Robots.php (accessed May 2010).

Ohishi, Kiyoshi, and Hiroshi Ohde. "Collision and Force Control for Robot Manipulator without Force Sensor." *IEEE International Conference on Industrial Electronics, Control and Instrumentation.* Bologna: IEEE, 1994. 766-771.

Olsen, M M, and H G Petersen. "A New Method of Estimating Parameters of a Dynamic Robot Model." *IEEE Transations on Robots and Automation*, 2001: 95-100.

Papakostas, Thomas V, Julian Lima, and Mark Lowe. "5.3: A Large Area Force Sensor for Smart Skin Applications." *IEEE Sensors*, 2002: 1620-1624.

Paul, R P, and B Shimano. "Kinematic Control Equations for Simple Manipulators." *IEEE Conference on Decision and Control.* 1978.

Peratech. *Touch Sensitive Skin Research Project for Robotic Applications: DefenseFile.* February 23, 2010. http://www.defensefile.com/Customisation/News/Education_Training_and_Professional_Services/Academic_Institutions_Higher_Education_Research/Touch_Sensitive_Skin_Research_Project_for_Robotic_Applications.asp (accessed May 2010).

Pholsiri, Chalongrath, Dinesh Rabindran, Mitch Pryor, and Chetan Kapoor. "Extended Generalized Impedance Control for Redundant Manipulators." *IEEE International Conference on Decision and Control.* Maui, Hawaii: IEEE, 2003. 3331-3336.

RAD. *Compliance Devices-Spring Loaded: RAD.* 2010. http://www.rad-ra.com/Compliance-Devices-Spring-Loaded.htm (accessed May 2010).

Ralph, Scott K, and Dinesh K Pai. "Detection and Localization of Unmodeled Manipulator Collisions." *IEEE International Conference on Intelligent Robots and Systems.* Pittsburg, Pennsylvania, 1995. 504-509.

RobotWorx. *Intelligent Robot incorporates force and vision sensors: RobotWorx.* September 27, 2004. http://www.robots.com/articles.php?tag=3213 (accessed May 2010).

Saito, Naoki, Shinya Kajikawa, and Hideharu Okano. "Development of a Flexible Structural Contact Force Sensor for a Grasping Task by Robot Hand." *IEEE International Conference on Industrial Technology.* 2003. 123-128.

Salisbury, J Kenneth. "Active Stiffness Control of a Manipulator in Cartesian Coordinates." *IEEE International Conference on Decision and Control including the Symposium on Adaptive Processes*, 1980: 95-100.

Sandia National Laboratories. *'Mighty Mouse' robot frees stuck radiation source.* December 14, 2005. http://www.sandia.gov/news-center/news-releases/2005/manuf-tech-robotics/mm-robot.html (accessed May 2010).

Seraji, Homayoun, David Lim, and Robert Steele. "Experiments in Contact Control." *Journal of Robotic Systems*, 1996: 53-73.

Shadow Robot Company Ltd. *Latest Hand News.* http://www.shadowrobot.com/hand/ (accessed May 2010).

Son, Jae S, Eduardo A Monteverde, and Robert D Howe. "A Tactile Sensor for Localizing Transient Events in Manipulation." *IEEE International Conference on Robotics and Automation*, 1994: 471-476.

Stilman, Mike. "Global Manipulation Planning in Robot Joint Space with Task Constraints." *IEEE Transactions on Robotics*, 2010: 576-584.

Swint, Ethan Baggett, and Del Tesar. *Collision Detection and Obstacle Avoidance for Robotic Manipulators.* Masters Thesis, Austin: University of Texas at Austin, 2004.

Takakura, Shinji, Toshiyuki Murakami, and Kouhei Ohnishi. "An Approach to Collision Detection and Recovery Motion in Industrial Robot." *Annual Conference of IEEE Industrial Electronics Society.* Philadelphia, PA: IEEE, 1989. 421-426.

Thomas, M, and Del Tesar. "Dynamic Modeling of Serial Manipulator Arms." *ASME Journal of Dynamic Systems and Control*, 1982: 218-228.

Turner, Cameron, and Jane Lloyd. *Actinide Research Quarterly--Automating ARIES.* July 2008. http://www.lanl.gov/source/orgs/nmt/nmtdo/AQarchive/1st_2ndQuarter08/page8.shtml (accessed May 2010).

Uchiyama, Masaru, and Kosei Kitagaki. "Dynamic Force Sensing for High-Speed Robot Manipulation Using Kalman Filtering Techniques." *Conference on Decision and Control.* Tampa, Florida, 1989. 2147-2152.

Willow Garage, inc. *PR2 Overview: Willow Garage.* http://www.willowgarage.com/pages/robots/pr2-overview (accessed May 2010).

Xia, Jinjun, Zongwu Xie, Honggen Fang, Tian Lan, Jianbin Huang, and Hong Liu. "Collision Detection of Flexible Joint Manipulator by Using Joint Torque Sensors." *IEEE/ASME International Conference on Advanced Intelligent Mechatronics.* Singapore: IEEE/ASME, 2009. 1816-1821.

Xie, H P, R V Patel, S Kalaycioglu, and H Asmer. "Real-Time Collision Avoidance for a Redundant Manipulator in an Unstructured Environment." *IEEE International Conference on Intelligent Robots and Systems.* Victoria, BC, Canada: IEEE, 1998. 1925-1930.

Yoshikawa, Tsuneo. "Force Control of Robot Manipulators." *IEEE International Conference on Robotics and Automation.* San Francisco, CA: IEEE, 2000. 220-226.

Zeng, Ganwen, and Ahmad Hemami. "An Overview of Robot Force Control." *Robotica*, 1997: 473-482.

Zhao, Y, C C Cheah, and JJ E Slotine. "Adaptive Jacobian Motion and Force Tracking Control for Constrained Robots with Uncertainties." *IEEE International Conference on Robotics and Automation.* Orlando, FL: 2006, 2006. 2226-2231.

Zheng, Yuan F, and Fred R Sias. "Two Robot Arms in Assembly." *IEEE International Conference on Robotics and Automation.* 1986. 1230-1235.

Vita

Kyle Schroeder was born in Toledo, Ohio on 1 April 1983. He was raised in Ohio but made a brief, one-year trip to live in Michigan before returning to Toledo to attend St. John's Jesuit High School. The summer after graduation in 2001, Kyle attended Fast Track Calculus at Rose-Hulman Institute of Technology wherein he completed all of the requirements for calculus in an intense 5-week course. At Rose-Hulman, Kyle was a DJ at the college radio station and worked as a lifeguard at the recreation center. He took no courses the winter and spring terms of his third year in order to do a co-op at The JDI Group, an architectural and engineering consulting firm in Holland, Ohio. During his fourth and final year at Rose, Kyle was a Residents' Assistant.

After graduating with his Bachelor's of Science from Rose-Hulman, Kyle worked at Continental Tire in Hannover, Germany. For about 1 year, he participated in an international trainee pool before deciding it was time to return to the United States. In 2008, Kyle returned to school at The University of Texas at Austin. He is working towards a PhD in Mechanical Engineering so he can teach young, enthusiastic engineers. He hopes to demonstrate the same passion and vigor as some of his favorite and most influential teachers and professors.

This thesis was typed by the author.